

SOUTHWEST FISHERIES CENTER

LA JOLLA, CA 92038

P.O. BOX 271

SOUTHWEST FISHERIES CENTER

AERIAL MARINE RESOURCES
MONITORING SYSTEM

BY

JOSPEH P. CARUSO, CINDY A. MEYER
AND SUSAN IACOMETTI

LIBRARY

DEC 19 1983

National Marine Fisheries Service
2570 Dole Street
Honolulu, Hawaii

APRIL 1979

REVISED MARCH 1983

ADMINISTRATIVE REPORT LJ-79-09



Digitized by the Internet Archive
in 2025

SERIAL PAVING STONE OF THE TOWN OF NEW YORK

1. The first of the

2. The second of the

3. The third of the

4. The fourth of the

5. The fifth of the

6. The sixth of the

7. The seventh of the

8. The eighth of the

9. The ninth of the

10. The tenth of the

11. The eleventh of the

12. The twelfth of the

13. The thirteenth of the

"This report is used to insure prompt dissemination of preliminary results, interim reports, and special studies to the scientific community. The material is not ready for formal publication since the paper may later be published in a modified form to include more recent information or research results. Abstracting, citing, or reproduction of this information is not allowed. Contact author if additional information is required."

AERIAL MARINE RESOURCES MONITORING SYSTEM

Joseph P. Caruso

April 1979

National Oceanic and Atmospheric Administration
National Marine Fisheries Service
Southwest Fisheries Center
La Jolla, California 92038

Revised by

Cindy A. Meyer and Susan Iacometti

March 1983

National Oceanic and Atmospheric Administration
National Marine Fisheries Service
Southwest Fisheries Center
La Jolla, California 92038

TABLE OF CONTENTS

Introduction.....	2
Areas.....	3
Log recording.....	3
Species identification.....	3
Tonnage interpretation.....	4
System overview.....	4
Data coding.....	5
Data verification.....	6
Data base.....	6
Data Coding Manual.....	8
Usage:general.....	9
EFFORT.....	10
Effort record definition.....	10
Effort coding sheet.....	11
Coding form - effort.....	12
Hour meter reading.....	12
Flight path.....	12
Adjusted flight path.....	13
Ambiguous flight path.....	14
Flight direction.....	15
Day-night code.....	16
Comment code.....	16
Unknown effort.....	17
Overland flights.....	19
Flight logs.....	20
SIGHTING.....	26
Sighting record definition.....	26
Sighting coding sheet.....	28
Coding form - sighting.....	29
Description and examples of the sighting types.....	29
Type 1 sighting.....	29
Type 2 sighting.....	30
Type 3 sighting.....	31
Type 4 sighting.....	32
Type 5 sighting.....	33
Type 6 sighting.....	34
Tonnage/block assignment.....	35
School ranges.....	36
Tonnage ranges.....	36
Individual school tonnages.....	36
Total tonnage.....	36
School/block assignment.....	37
Fractional tonnages.....	40
Individual school tonnages.....	40
Total tonnage.....	42
Combined species codes.....	43
Mammal sightings.....	45
Spots or bunches.....	46

Ambiguously drawn circles.....	46
Ambiguous sighting reports.....	46
Ambiguous classification.....	48
Comment code.....	49
Comment code - expanded.....	50
Red tide.....	51
Sonar-spotted school.....	51
Sighting type codes.....	52
Pilot reference table.....	53
Plane identification.....	54
Species code numbers.....	57
Species codes - supplemental list.....	62
Block numbering method.....	69
Block number conversion table.....	70
Table of all possible 6-digit block numbers....	77
Coded log examples.....	92
Data Editing.....	106
Subdirectories.....	106
File names.....	107
Form management system.....	108
Form descriptions.....	109
CODE - a data entry program.....	119
Program description.....	119
Program listing.....	120
Data editing summary.....	124
Using the editing programs.....	126
Editing programs.....	127
SORT.....	127
Program description.....	127
Command file listing.....	128
SDISK.....	130
Program description.....	130
Command file listing.....	131
Program listing.....	132
SEEDIT.....	135
Program description.....	135
Command file listing.....	138
Program listing.....	139
EDISK.....	145
Program description.....	145
Command file listing.....	146
Program listing.....	147
FMSEEDIT.....	151
Program description.....	151
Command file listing.....	152
Program listing.....	153
EEDIT.....	156
Program description.....	156
Command file listing.....	158
Program listing.....	159
MERGE.....	164
Program description.....	164
Command file listing.....	165
Program listing.....	166

Effort record as stored in the database.....	170
Effort records over 77 blocks long.....	171
Datatrieve.....	172
List of domains.....	173
Effort record description.....	174
Sighting record description.....	175
Domain creation.....	176
Tapes.....	177
Saveset listings.....	178
Data editing references.....	184
Yearly Summaries of Effort and Sighting	
Data Cards and Number of Flights.....	185
RECCNT - record counting program.....	196
Program description.....	196
Program listing.....	197
Change of record formats.....	198
VTOF - effort record format change program.....	199
Program description.....	199
Program listing.....	200
References.....	201
APPENDIX - Data usage example.....	202
The anchovy index.....	202
INDEX - the index calculation program.....	205

Aerial Marine Resources Monitoring System

INTRODUCTION

The pelagic fish aerial monitoring system began in 1962 under the direction of James Squire. Commercial fish spotters were contracted by Mr. Squire to record the location and abundance of pelagic fish schools. Emphasis was placed on fish species of commercial importance. The data from pilot observation logs are used to calculate an annual index of abundance for the important fish species. Squire (1983) published a summary of the data and indices for 1962 through 1978.

During the first year the fish spotters recorded their observations on tape recorders, and these tapes were later transcribed by National Marine Fisheries Service (NMFS) personnel. This recording method allowed for transcription errors to occur in the flight path and area of sightings. The next year this problem was eliminated by changing to the use of flight logs, with the pilots completing the logs themselves. These first logs were simple, depicting the coast and island outlines, and the pilots recorded their flight path and sightings freely. In 1968, the second generation of flight logs was brought into use. In addition to the coast and island outlines, the new logs included a grid system based on 10 min. squares.

Since the start of the aerial monitoring system, abundance had been calculated by hand. As the amount of data grew, it became apparent that faster and more efficient means of calculating the index were needed. In 1977, Joe Caruso was given the responsibility of setting up a system which made the data more accessible, thus facilitating calculations and investigations. A data base was created on the INFONET system of the Computer Science Corporation (CSC). In 1981, federal government institutions were not allowed to renew their contracts with CSC, and the existing data base was transferred to the Burroughs 7800 computer on the UCSD campus. Steps had already begun for transferring the entire data base and its programs to the VAX mini-computer at UCSD when notice was given in the beginning of 1982 that the Burroughs would no longer be available for anything but administrative usage after June 1982. Plans for acquiring a VAX for NMFS at the La Jolla Laboratory further strengthened the case for moving the data base. A data base management software package, DATATRIEVE, which is available on the VAX, provides access and report-writing capabilities.

The data is categorized into two groups, effort and

sighting. Special coding forms have been designed and printed for each. The coding manual insures consistency and uniformity throughout the coding process which is vital when dealing with time series data such as this. Coders are trained by someone who is already familiar with the steps involved. Data verification programs have been written. Each year of data is verified, corrected, and stored.

Data can be retrieved from the data base for the calculation of the index of abundance. Sighting power and fish interaction analysis can also be performed on the data. The data are available for any applicable analysis. A summary of the amount of data currently on the data bases is given in Appendix III.

AREAS

The pilots normally fly out of Santa Barbara, Los Angeles, and San Diego areas. They survey a region that extends from Half Moon Bay to Cedros Island (off of Mexico) and as far offshore as Tanner and Cortez Banks. The majority of flights are concentrated in the Southern California Bight extending from Pt. Conception to Ensenada, Baja California, where the southern California purse seine fleet expends most of its effort.

LOG RECORDING

The accuracy of a pilot's flight path and sighting varies. The pilots fly year-round, day and night, depending on the condition of their aircrafts, the weather, and the fleet activity. Flight logs are completed after landing because the pilots are too busy when flying, and at night it is too dark. Recording accuracy probably increases the closer the pilot's observations are to identifiable landmarks.

SPECIES IDENTIFICATION

School size, shape, and speed are but a few ways in which pilots identify fish species (Squire, 1972). Each pilot has his own identification techniques. The fishing fleet depends upon the pilots to make correct identifications. The captains of these vessels rely on the pilots and respond when they report a particular species in the area.

TONNAGE INTERPRETATIONS

Questions concerning accuracy and precision arise when the pilots report tonnages for the sightings. Generally the pilots can see only two dimensions of any school of fish. Considerations such as the density of the school, depth of the school in the water column, phosphorescence in the water, water clarity, amount of sunlight and fish dispersion can make observations of the total school extremely difficult.

Accurate reporting of school tonnages is made in one of two ways: 1) the pilots can estimate the entire school or 2) the pilot's estimation is made in conjunction with the ship captain's catch estimate. The first is self-explanatory. The second estimation is made after the ship's captain reports the tonnage and percentage of the school he caught.

Precision of the pilots' observation in reporting school tonnages was analyzed by MacCall (1975). He found consistent relative differences in reported school tonnages between pilots. Also, MacCall observed that relative differences must be accounted for in future calculations of abundance indices for various species.

SYSTEM OVERVIEW

The Aerial Marine Resources Monitoring System (AMRMS) is a series of sequential activities. It starts with the pilots recording what they see on flight logs. The data is then coded, and directly entered onto disk storage on the VAX system where it is verified. When all verifications procedures are completed and the data are corrected, the data is stored (and backed up) on magnetic tape. The data can then be analyzed providing information of various types.

So that the reader can visualize the process involved, an outline of the activities follows.

- * Receive logs from pilots
- * Code logs on code sheets
- * Key data onto disk from coding sheets
- * Verification programs:
 - Effort data --
 - 1) data to disk
 - 2) sort
 - 3) translate block from three- to six-digit numbers

- 4) verify the data
- 5) correct errors
- 6) re-verify the data

-- Sighting data --

- 1) data to disk
- 2) translate block from three- to six-digit numbers
- 3) sort
- 4) verify the data
- 5) correct errors
- 6) re-verify the data

-- Effort and Sighting data --

- 1) merge and cross-check

- * Store (and back up) verified data on tape
- * Analyze data

Two data bases are maintained, one for the effort data and the other for sighting. The primary data analysis will be the estimation of annual indices of abundance for major pelagic schooling fish species in the Southern California Bight. Other types of descriptive analyses, such as tonnage and schooling data by species and block area can be performed and are done so upon request. As of 1982, the AMRMS data base utilization includes providing information used in the anchovy biomass estimation.

DATA CODING

The flight log information is keypunched in two groups, effort and sighting. The effort and sighting coding forms have been designed for ease of coding and keypunching the data collected.

Effort is defined as the number of ocean blocks over which a pilot passes. The effort data contain the approximate flight path, in consecutive blocks, of each pilot for each flight. This method was adopted to preserve as much information from the logs as possible.

A sighting is defined as a drawn circle describing the limits of a group of fish or mammals on a pilot's flight log. The sighting data, for the most part, consists of the specific block area, the species involved, the tonnage, and the number of schools as seen by the pilot. A sighting which overlaps two or more block areas may be split into two or more separate sightings. Interpretations and definitions are given in the Data Coding Manual in the next section.

DATA VERIFICATION

The AMRMS source data are initially processed by verification programs to insure data accuracy. By correcting errors at the onset, cost savings in time and money are realized by minimizing or eliminating data corrections on data base files during any analysis.

The data are verified in three stages: effort data alone, sighting data alone, and effort and sighting data together. These checks provide adequate error coverage, both usual and unusual, and only very subtle errors pass by undetected. When the data have been completely verified and corrected, it is written to tape (and backed up) for storage and ready for analysis.

Verification is achieved by checking specific ranges or known quantities against the input variables. Some variables which cannot be verified are: flight path accuracy, area sighting accuracy, and tonnage accuracy. In other words, we cannot check the pilots' recordings. We can and do check for keypunch and coding errors whenever possible. A complete list of error checks are given in the Program Overview sections for effort and sighting records in the third section of this manual.

DATA BASE

A data base is a computer system for easy access and retrieval of data. Two data bases have been designed for the AMRMS, using DATATRIEVE data base management system on the VAX mini-computer.

The first data base contains the effort data records. An effort record contains the year, month, day, pilot, flight, time, plane, and block numbers for all blocks covered on the flight. The second base has the sighting data. The sighting records have the same first seven elements as the effort records (columns 1-11 containing the year, month, day, pilot, flight, time, and plane) and, in addition, the block number where the sighting occurred, species code, schooling, and tonnage information.

Each data base is a unique set of records; the two data bases are interrelated by their first elements. In this way as much information as possible can be obtained from each flight. New verified data may be added to the data bases by year, or smaller sets may be added at any time.

Data bases have advantages over disk or tape files for

a number of reasons:

- 1) The software is already written for you.
- 2) With the software written, users may include non-programmers.
- 3) Large data records are easily handled.
- 4) Each record is uniquely keyed and may contain secondary keys (for sorting, etc.).
- 5) Each record is quickly accessible because of keys.
- 6) Report writing capabilities have been included in the software.

Although data bases are expensive to create and set up, they are relatively inexpensive to maintain.

AERIAL MARINE RESOURCE MONITORING SYSTEM

DATA CODING MANUAL:

Explanation and Examples

of the

Coding Forms

for the

Effort and Sighting Records

General and specific usage notes have been provided to explain certain details of coding which are not obvious. These notes will serve as a guide by which the coder may adequately deal with certain coding situations and solve coding problems.

Usage: General

- 1) The data are coded on formats made especially for these records.
- 2) All coded entries are numeric.
- 3) Each field is right-adjusted.
- 4) Leading zeroes are not necessary and may be omitted.
- 5) The coder will enter, in the appropriate field, the 3- or 6-digit block number found on the flight log or overlay. An editing computer program will change all 3-digit block numbers to the revised 6-digit block number.
- 6) Each coding form provides enough space for two average flight logs.
- 7) Completed coding forms should be stacked in homogeneous groups.
- 8) Data entry will occur directly from these special forms onto the disk of the VAX mini-computer used by the Fisheries Center personnel.

Definitions

AERIAL MARINE RESOURCE MONITORING SYSTEM

- EFFORT RECORDS -

<u>Column</u>	<u>Title</u>	<u>Explanation</u>
1-2 3-4 5-6	Date { Year Month Day	The date of each flight, broken down into the year (2 digits), month (2 digits) and day (2 digits). 6 digits.
7-10	Pilot Code	The name of the pilot, in code, for purpose of security and formatting ease. 4 digits.
11	Flight No.	The sequential flight # made by one pilot in a 24-hour period. 1 digit.
12	Day/Night Code	This code distinguishes between daylight and nighttime flights. Day = 1, Night = 2. 1 digit.
13-19	Tach Time Out	The hour meter reading at the time of take-off. 7 digits (one-place implied decimal).
20-26	Tach Time In	The hour meter reading at the end of the flight. 7 digits (one-place implied decimal).
27-29	Plane Code	The identification number of each plane will have a corresponding code for security and formatting ease. 3 digits.
30	Comments	Lets the user know if the pilot made any comments on his aerial survey log sheet. Yes = 1, No = 2. 1 digit.
31-36	Block No. (1)	Represents the latitude (3 digits) and longitude (3 digits) of a ten minute square block of ocean searched by the pilot. The latitude and longitude of the southeastern corner of the block defines its entirety. The blocks are ordered according to the flight path. Search is defined as a block flown through, irrelevant of the time spent in that block. 6 digits (1st 3 digits of block # are latitude in minutes divided by 10; 2nd 3 digits are longitude divided by 10).
37-42	Block No. (2)	
.	.	
.	.	
73-78	Block No. (8)	
79-80	Sequence No.	Indicates the sequence of the effort card in the flight. 2 digits.

Usage: Effort

Coding Form - Effort

Each page of the Aerial Marine Resource Monitoring System Effort records contains two distinct coding forms. The coder should be able to record, on the average, the effort from one flight log on each form. If, however, the coder finds that the information contained on one flight log is too much to fit on one form, the coder will continue the data coding onto the next form. In this case, the repeated elements (Columns 1-30 inclusive) are coded exactly the same on the second form as on the first form.

Hour Meter Reading

The Tach Time In and the Tach Time Out are the only floating point numbers in this format. One decimal place is reserved on the coding form for each field. If the pilot records his hour meter reading to the nearest hundredth of an hour, the coder will round off the time to the nearest tenth of an hour before recording the data. If the hundredths digit is a five, rounding occurs to the nearest even tenth of a whole number. If the hundredth digit is not a five, rounding occurs to the nearest tenth of a whole number.

Example:

Pilot Time out 3475.25
Records Time in 3481.33

Coded Example:

HOUR METER READING																									
TIME OUT													TIME IN												
13	14	15	16	17	18	19	20	21	22	23	24	25	26												

Flight Path

The block area numbers are coded according to the flight path shown on the flight log as interpreted by the coder.

Usage: Effort (continued)

Ambiguous Flight Path

If a pilot records a flight path that travels along the edge of a block for at least one-half of a block length, it will be assumed that he searched all of the blocks in between which he flew.

Flight Log Example:



Coded Example:

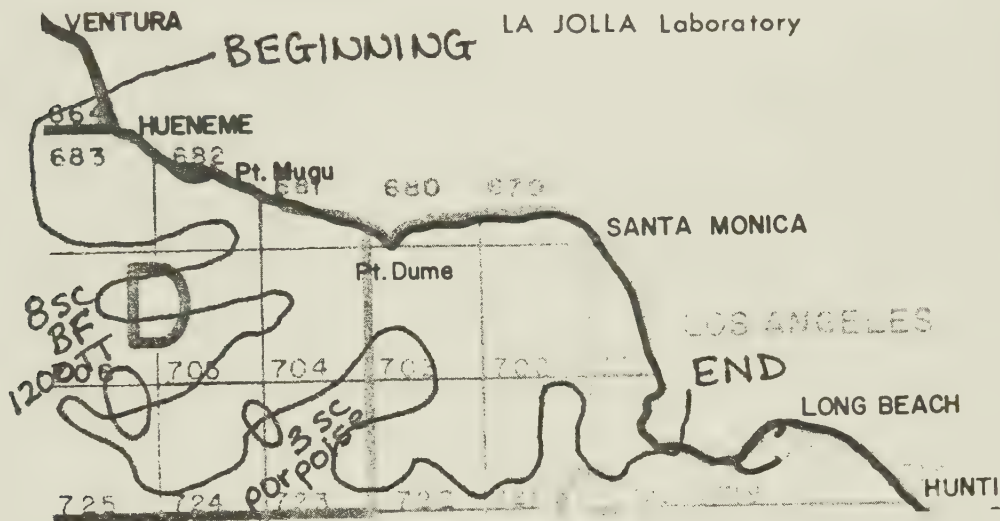
BLOCK NUMBERS SEQUENCED ACCORDING TO THE INTERPRETED FLIGHT PATH								Sequence
842	843	844	822	823	801			1
32 33 34 35 36	37 38 39 40 41 42	43 44 45 46 47 48	49 50 51 52 53 54	55 56 57 58 59 60	61 62 63 64 65 66	67 68 69 70 71 72	73 74 75 76 77 78	79 80

Usage: Effort (continued)

Flight Direction

The direction of the flight path will be defined as beginning on the northern or westernmost part of the flight path, as drawn on the flight log, and ending on the southern or easternmost part of the flight path, unless otherwise indicated.

Example:



Usage: Effort (continued)

Day-Night Code

The Day-Night Code consists of a code for daylight flights (1), a code for nighttime flights (2), and a separate code (3) for flights that incorporate both or when the pilot does not indicate where the nighttime part of the flight ended and the daytime began or vice-versa.

Comment Code

The comment code is used to record any written information not normally coded that the pilot reports on the flight log. Some comments will be general in nature and pertain to the whole flight. If this is the case, the coder will record a positive comment in the effort record. A positive comment will be recorded if there were any sonar-spotted schools of fish on that flight log. A positive comment will be recorded if there was any red tide on the flight log.

<u>Code</u>	<u>Comments</u>
1	Yes
2	No

Example I:

Flight Log Example:

639	638	637	Foggy Day

Pt. Arguello

GAVIO

Coded Example:

Comments
1
30

Usage: Effort (continued)

Example II:

Flight Log Example:



Coded Example:

Comments
I
30

Unknown Effort

If a pilot flies into an area for which he does not have a flight log, his effort for that part of the flight will be coded as unknown. Special three (3) digit codes have been assigned to the flight logs and incorporated into the coding to tell us into which area the unknown effort took place, as well as to preserve a non-bias account of the flight. The special codes are:

- 1) Flight Log I-991; 2) Log II-992; 3) Log III-993;
4) Log IV-994; 5) Log V-995.

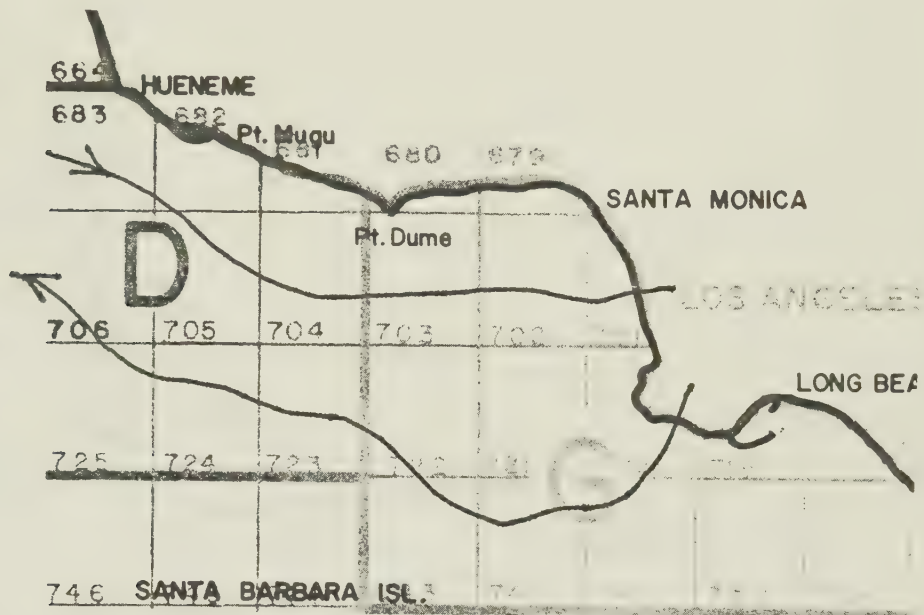
Sightings that are recorded by the pilot in unknown areas will be coded normally, with the exception that the block area in which the sighting took place will be coded as a 991, 992, 993, 994, or 995.

Usage: Effort (continued)

Example:

A pilot flies out of Long Beach up to Santa Barbara and back again. The only flight log he has is Log III, which covers the Long Beach area.

FLIGHT LOG EXAMPLE:



CODED EXAMPLE:

BLOCK NUMBERS SEQUENCED ACCORDING TO THE INTERPRETED FLIGHT PATH								Sequence
720	741	742	743	722	723	724	725	1
706	993	683	682	705	704	703	702	2
701								3
32 33 34 35 36	37 38 39 40 41 42	43 44 45 46 47 48	49 50 51 52 53 54	55 56 57 58 59 60	61 62 63 64 65 66	67 68 69 70 71 72	73 74 75 76 77 78	79 80

If a pilot flies over land for a part of the flight, he is not searching for that part of the flight. A special three-digit code has been assigned to tell us that the pilot was not searching because he was over land. The special code is 999.

The map shows the coastline of Chile and Argentina. Latitude lines are marked at 32° 30' S, 33° 00' S, 33° 30' S, 34° 00' S, and 34° 30' S. Longitude lines are marked at 71° 30' W, 72° 00' W, 72° 30' W, 73° 00' W, 73° 30' W, 74° 00' W, 74° 30' W, 75° 00' W, and 75° 30' W. Key locations marked are Pt. Sal, Pt. Arguello, Pt. Concepcion, Gaviota, Capitan, and Goleta. A north arrow is present.

BLOCK NUMBERS SEQUENCED ACCORDING TO THE INTERPRETED FLIGHT PATH																								Sequence																								
654				655				656				999				637				638				632				631				1																
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80

Usage: Effort (continued)

Flight Logs

The following logs are the type of flight logs the spotter pilots complete. Refer to page 92 for sample coded logs.

Usage: Effort (continued)

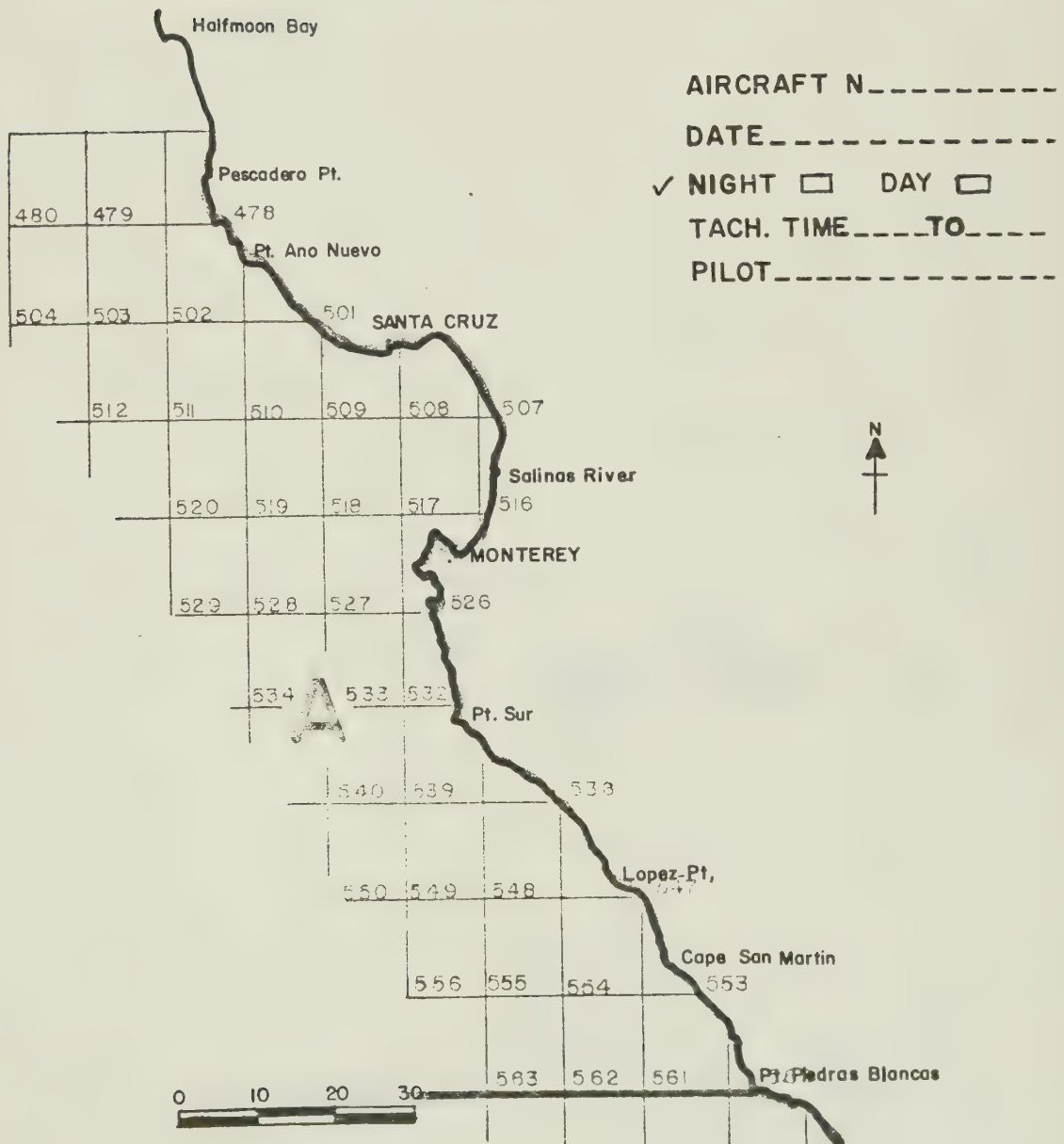
LOG I - 991



U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration

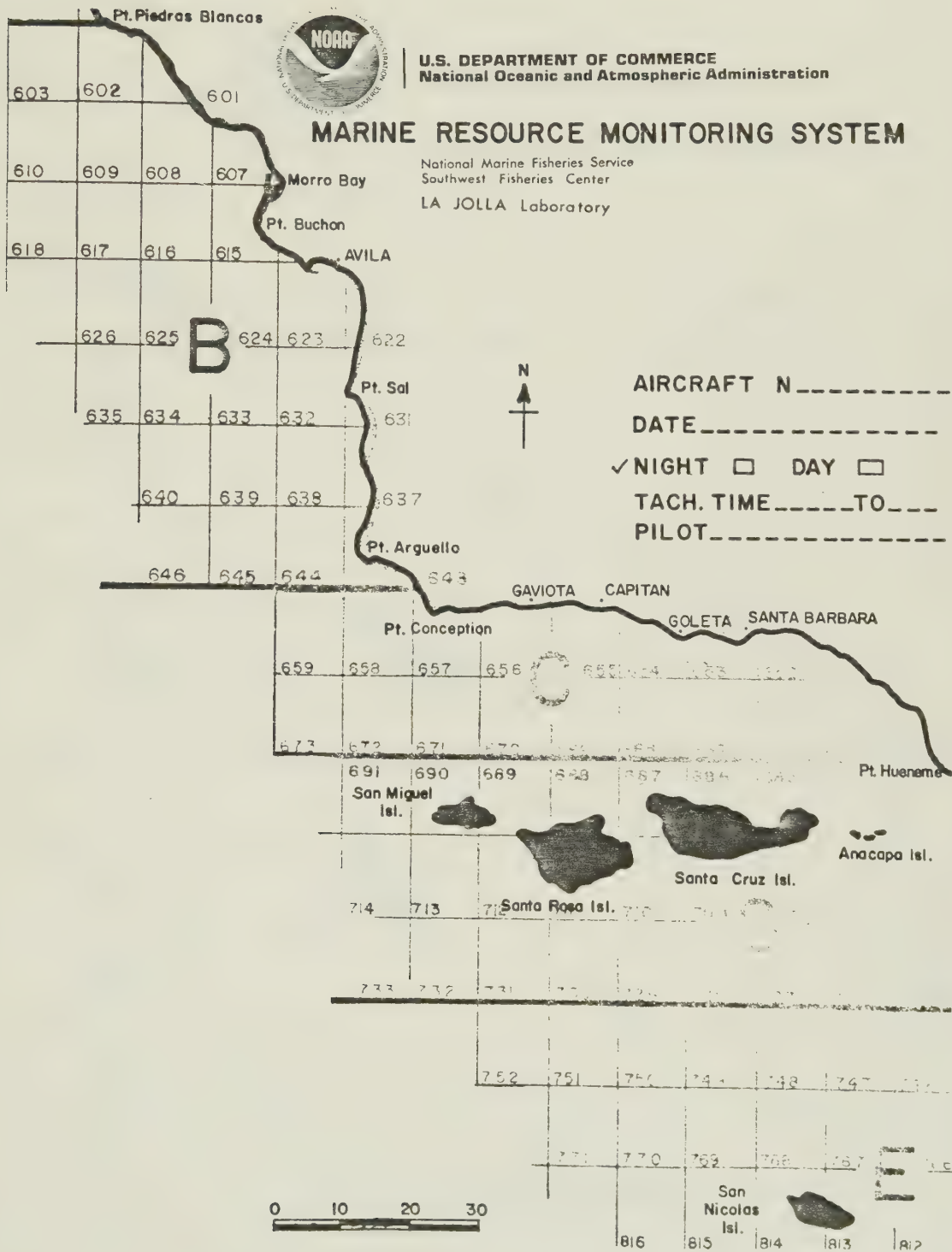
MARINE RESOURCE MONITORING SYSTEM

National Marine Fisheries Service
Southwest Fisheries Center
LA JOLLA Laboratory



Usage: Effort (continued)

LOG II - 992



Usage: Effort (continued)

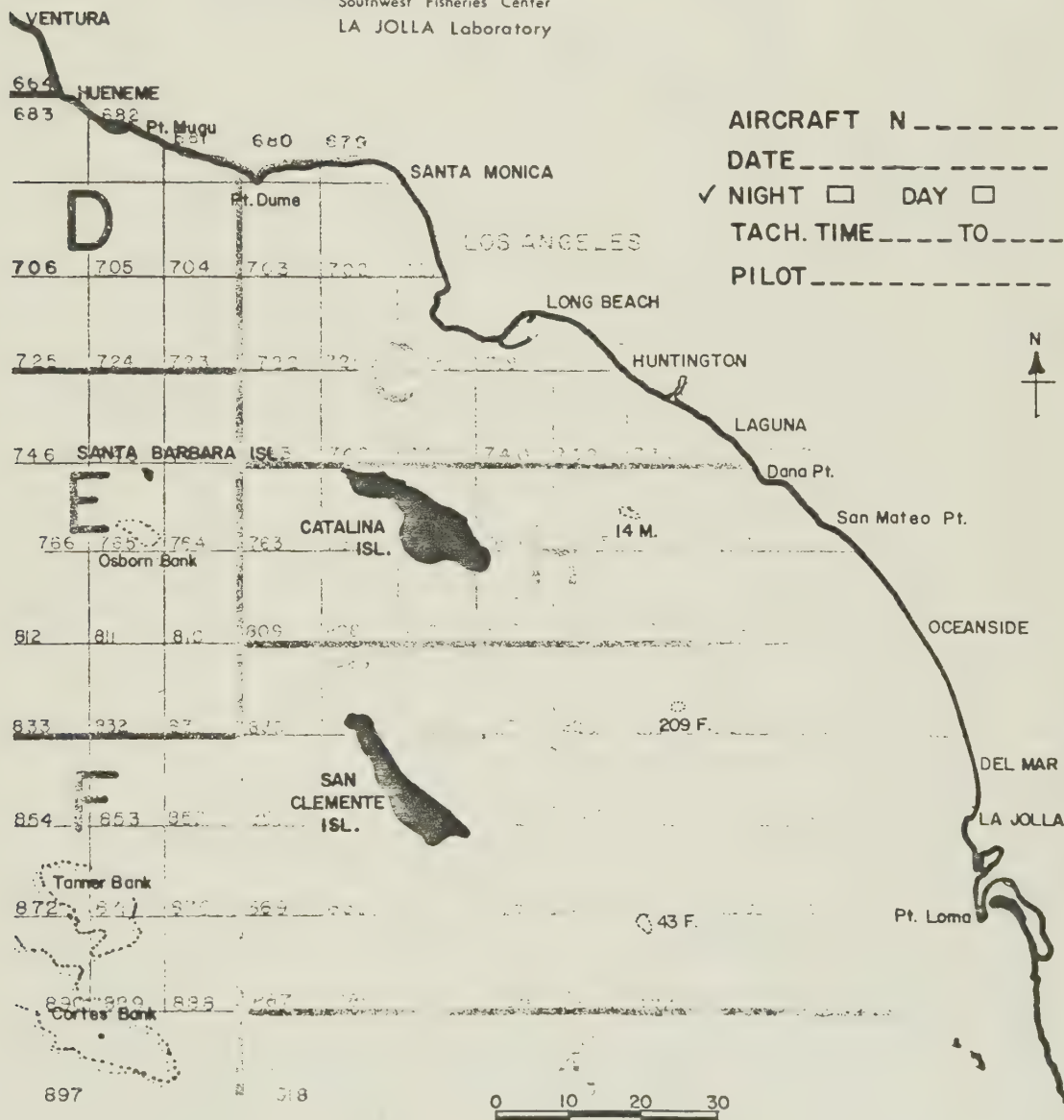
LOG III - 993



U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration

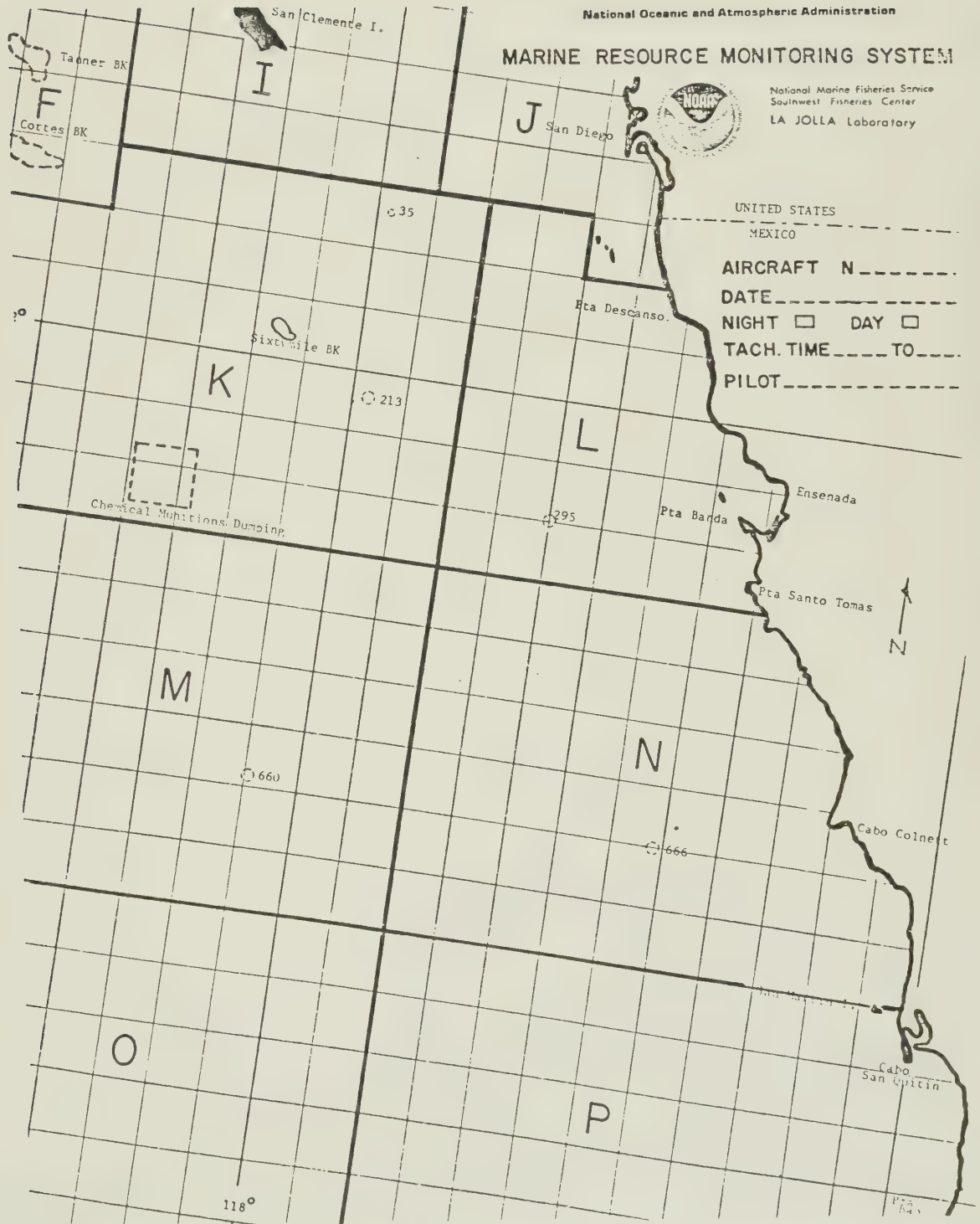
MARINE RESOURCE MONITORING SYSTEM

National Marine Fisheries Service
Southwest Fisheries Center
LA JOLLA Laboratory



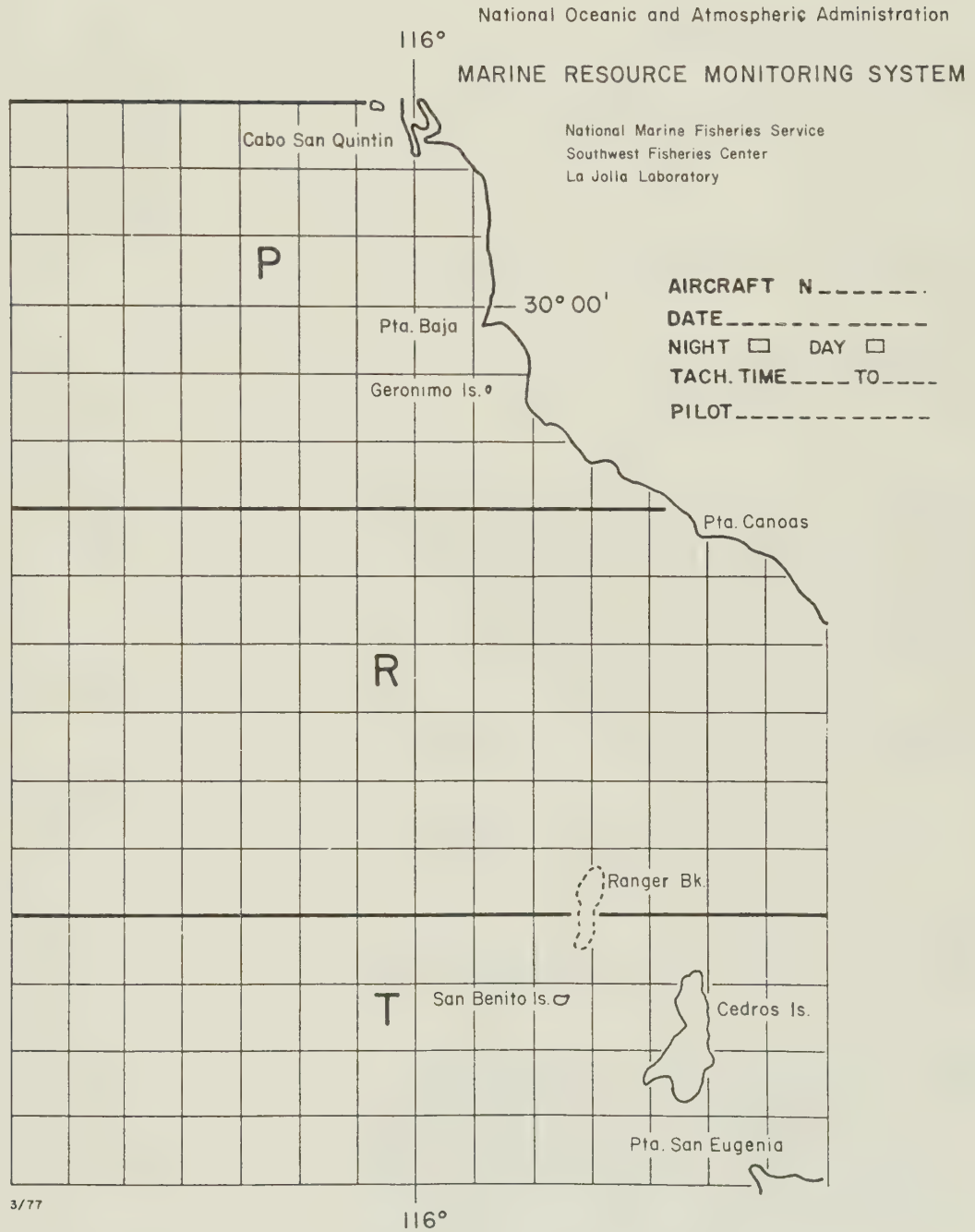
Usage: Effort (continued)

LOG IV - 994



Usage: Effort (continued)

LOG V - 995



Definitions

AERIAL MARINE RESOURCE MONITORING SYSTEM

- SIGHTING RECORDS -

<u>Column</u>	<u>Title</u>	<u>Explanation</u>
1-2 3-4 5-6	Date { Year Month Day	The date of each flight, broken down into the year (2 digits), month (2 digits) and day (2 digits). 6 digits.
7-10	Pilot Code	The name of the pilot, in code, for purpose of security and formatting ease. 4 digits.
11	Flight No.	The sequential flight # made by one pilot in a 24-hour period. 1 digit.
12-13	Blanks	Fillers. 2 digits.
14-19	Block No.	Represents the latitude (3 digits) and longitude (3 digits) of a 10-minute square area of ocean in which a sighting was recorded. A sighting is defined as a school or schools of fish reported by the pilot in this area. 6 digits.
20-23	CF&G Species Code	The code number corresponding to the particular species of fish sighted. 4 digits.
24	Day/Night Code	This code distinguishes between daylight and nighttime flights. Day = 1, Night = 2. 1 digit.
25	Comments	Lets the user know if the pilot made any comments on his aerial survey log sheet. Yes = 1, No = 2. 1 digit.
26-27	Sighting Type Code	Classifies the sightings into distinct categories: 1) # of schools; 2) total tonnage; 3) # of individuals; 4) # of schools and total tonnage; 5) # of schools and individual tonnages for each school totalled; 6) no information other than species code. 2 digits.
28-31	Number of schools.	The number of schools sighted. 4 digits.
32-40	Total tonnage.	The total tons sighted. 9 digits.

- | | | |
|-------|------------------------|---|
| 41-44 | Number of individuals. | The number of individuals sighted.
4 digits. |
| 45 | Species table number. | References table on which the species
code numbers are found. 1 digit. |

Date: _____

DATE						PILOT					flight
YEAR		MONTH		DAY							
1	2	3	4	5	6	7	8	9	10	11	

[illegible]

DATE						PILOT					Flight
YEAR		MONTH		DAY							
1	2	3	4	5	6	7	8	9	10	11	

[illegible]

Usage: Sighting

Coding Form - Sighting

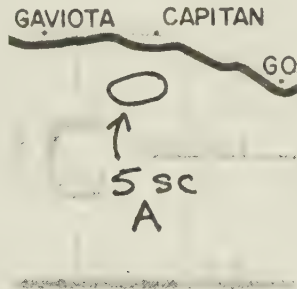
Each page of the Aerial Marine Resource Monitoring System Sighting records contains two distinct coding forms. The coder should be able to record, on the average, the sightings from one flight log on each form. If, however, the coder finds that there are more sightings to record than are allowed on one form, the data coding should be continued onto the next form. For such a case, the repeated elements on the second form (Columns 1-11 inclusive) are coded exactly the same as on the first form.

Description and Examples of the Sighting Types

TYPE 1 SIGHTING.

The pilot recorded an estimated number of schools of one species of fish in one block.

Flight log example:



Coded example:

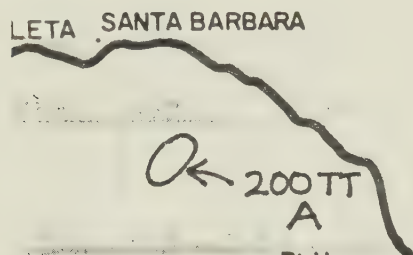
BLOCK NUMBER	CF & G SPECIES CODE	Day Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO																							
655	11022			1	5			1																							
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45

Usage: Sighting (continued)

TYPE 2 SIGHTING.

The pilot recorded a total tonnage for one species of fish in one block.

Flight log example:



Coded example:

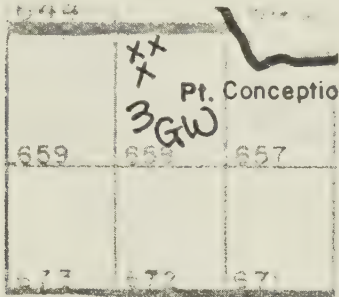
BLOCK NUMBER					CF & G SPECIES CODE					Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS					TOTAL TONNAGE										NUMBER OF INDIVIDUALS					SPECIES TABLE NO
666					11022					22		2						200															1
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45		

Usage: Sighting (continued)

TYPE 3 SIGHTING.

The pilot recorded an estimated number of individuals of one species of fish in one block.

Flight log example:



Coded example:

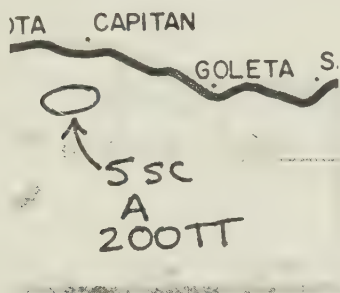
BLOCK NUMBER					CF & G SPECIES CODE					Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS					TOTAL TONNAGE										NUMBER OF INDIVIDUALS					SPECIES TABLE NO
658					2001					12		3																3					1
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45		

Usage: Sighting (continued)

TYPE 4 SIGHTING.

The pilot recorded an estimated number of schools and a total tonnage for the entire number of schools of one species of fish in one block.

Flight log example:



Coded example:

BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO
655	11022	4		5		200		1
14 15 16 17 18 19	20 21 22 23	24 25	26 27	28 29 30 31	32 33 34 35 36 37 38 39 40	41 42 43 44	45	

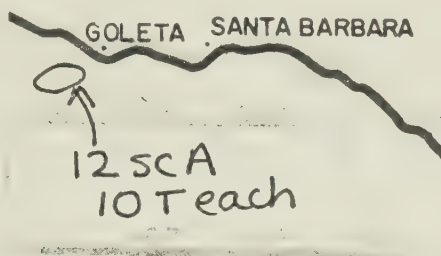
Usage: Sighting (continued)

TYPE 5 SIGHTING.

The pilot recorded an estimated number of schools of one species of fish and the individual tonnage for each school in one block (Example 1). Like tonnages will be totalled on the same record, with a new record created for each tonnage value (Example 2).

Example 1:

Flight log example 1:

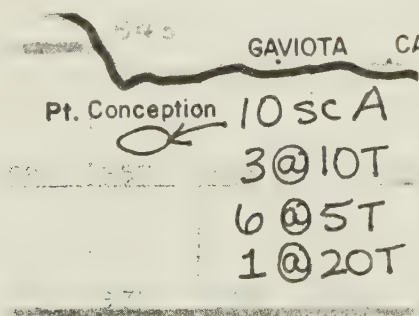


Coded example 1:

BLOCK NUMBER					CF & G SPECIES CODE					Day/Night		Comments		SIGHTING TYPE		NUMBER OF SCHOOLS					TOTAL TONNAGE										NUMBER OF INDIVIDUALS					SPECIES TABLE NO	
654					11022					22				5		12					120															1	
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45						

Example 2:

Flight log example 2:



Coded example 2:

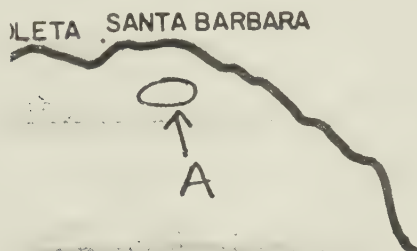
BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO																							
657	110	22		5	3	30		1																							
657	110	22		5	6	30		1																							
657	110	22		5	1	20		1																							
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45

Usage: Sighting (continued)

TYPE 6 SIGHTING.

The pilot only recorded a species name for one species of fish in one block.

Flight log example:



Coded example:

BLOCK NUMBER					CF & G SPECIES CODE					Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS					TOTAL TONNAGE												NUMBER OF INDIVIDUALS					SPECIES TABLE NO.
652					11022							6																							1
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45				

Usage: Sighting (continued)

Tonnage/Block Assignment

If a Type 2 sighting was recorded by a pilot and that sighting falls in more than one block area, the coder will break down the sighting into proportionate tonnages and record them in the block area involved.

For example, a pilot records 2,000 tons of bluefin tuna partially covering the block areas 845, 844, 862, and 863. The coder will then estimate the proportion of the drawn school located in each block area, say 50%, 25%, 10%, and 15%, respectively.

Flight log example:

Q 43 F.  Pt. Lc
2000 T
BF

Sightings of Type 2 of bluefin tuna would be coded in block areas 845, 844, 862, and 863 with tonnages equaling 1,000, 500, 200, and 300, respectively.

Coded Example:

BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO
845		4	1	2	2	1,000		1
844		4	1	2	2	500		1
862		4	1	2	2	200		1
863		4	1	2	2	300		1
14 15 16 17 18 19	20 21 22 23	24	25	26 27	28 29 30 31	32 33 34 35 36 37 38 39 40	41 42 43 44	45

Usage: Sighting (continued)

School Ranges

If a pilot reports a range of schools for a sighting, it will be assumed that the midpoint of the range is the actual number of schools sighted. Should rounding become necessary, simply round to the nearest even whole number.

Tonnage Ranges

A high and low value reported for the individual or total school tonnage is defined as the tonnage range. Tonnage ranges are handled differently for individual school or total tonnages.

A. Individual School Tonnages - If a pilot records a tonnage range for individual schools, the coder will calculate the average tonnage per school, multiply the average tonnage by the number of schools, round to the nearest whole number and record the product as the total tonnage.

B. Total Tonnage - If a pilot records a tonnage range for total tonnage, the coder will simply record the calculated average tonnage as the total tonnage.

Example:

A) Block area - 845

12 schools BF 10-15T each

B) Block area - 844

1,000-2,000 TT YF

. 11

Coded Example:

BLOCK NUMBER					CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE										NUMBER OF INDIVIDUALS	SPECIES TABLE NO.
845					4	12	5	12	150											1	
844					1	12	2		1500											1	

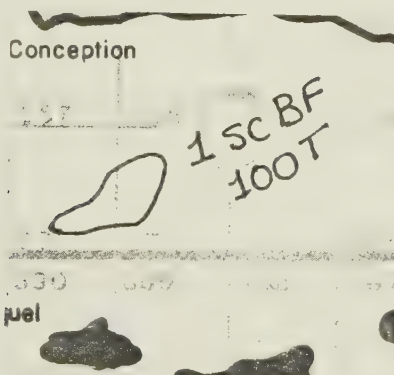
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

School/Block Assignment

A. If one school of fish is sighted and the pilot indicates that the school is located in more than one block area, the entire school will be recorded in the block where the largest part of the school is indicated (Example I). If the coder determines that the school indicated is evenly divided among the blocks into which it falls, the school will be assigned to the block into which the plane first entered, as indicated by the pilot on the flight log (Example II).

EXAMPLE I

Flight log example:



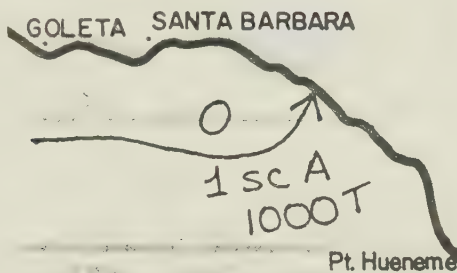
Usage: Sighting (continued)

Coded example:

BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO
670	412			5	1	1.00		1
14 15 16 17 18 19	20 21 22 23	24 25	26 27	28 29 30 31	32 33 34 35 36 37 38 39 40	41 42 43 44	45	

EXAMPLE II

Flight log example:



Coded example:

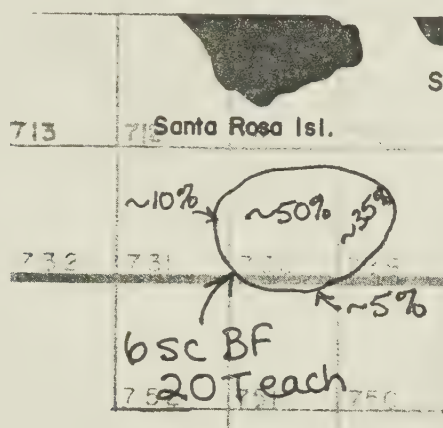
BLOCK NUMBER						CF & G SPECIES CODE		Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS		TOTAL TONNAGE								NUMBER OF INDIVIDUALS				SPECIES TABLE NO						
666						11012				5	1		1.000												1						
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45

Usage: Sighting (continued)

B. If a pilot sights a number of schools of fish and indicates that the sighting is located in more than one block area, the coder normally breaks down the drawn sighting, proportionately, by block area, and assigns one or more schools to the block areas involved. However, when the proportion of the drawn sighting in one block area multiplied by the number of schools sighted is less than 1/2 of a school, no schools will be assigned to that block area. This point was assumed previously.

EXAMPLE III

Flight log example:



Coded example:

BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO																							
731	412	5	1	20	1																										
730	412	5	3	60	1																										
729	412	5	2	40	1																										
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45

Note in the example that Block # 751 will not have any schools assigned to it. Block # 731 will have one, 730 will have three, and 729 will have two.

Usage: Sighting (continued)

Fractional Tonnages

A. Individual School Tonnages - If a fractional tonnage for each school of fish is reported, the number of schools is multiplied by the tonnage and rounded (Example I). If a pilot reports a tonnage that is less than 1/2 of a ton for all schools, the coder will record the number of schools, and enter a positive response in the comments field (Example II).

EXAMPLE I

Flight log example:

771 770
7SC → 0
YF
1/3T @ 816

Coded example:

BLOCK NUMBER					CF & G SPECIES CODE					Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS					TOTAL TONNAGE										NUMBER OF INDIVIDUALS					SPECIES TABLE NO.
816										112		5	7															2					1
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45		

EXAMPLE II

Flight log example:

→ 0
15SCA
1/3T
San Mig Isl.

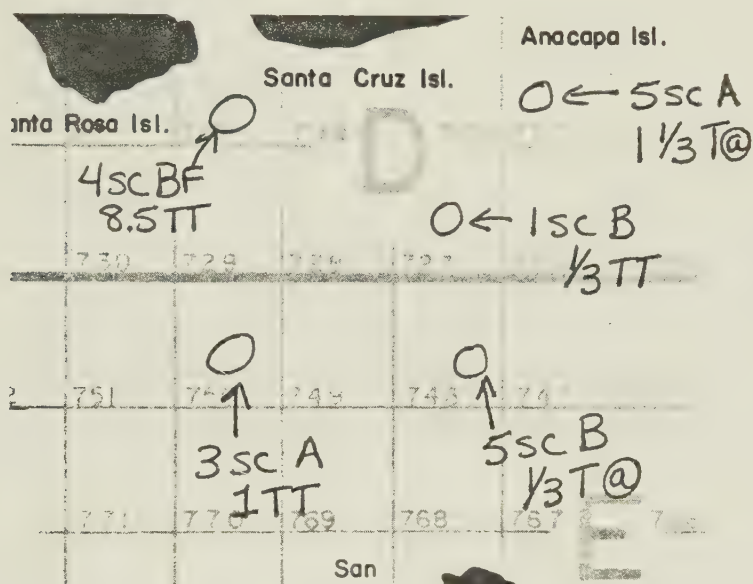
Usage: Sighting (continued)

Coded example:

BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO.
673	11011	1		1	15			1
14	15	16	17	18	19	20	21	22
23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40
41	42	43	44	45				

EXAMPLE III

Flight log example:



Coded example:

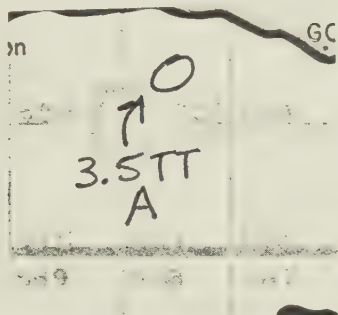
BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO.
710	412	4		4	4	9		1
707	11012	5		5	5	7		1
727	311	1		1	1			1
750	11012	4		3	1			1
748	312	5		5	2			1
14	15	16	17	18	19	20	21	22
23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40
41	42	43	44	45				

Usage: Sighting (continued)

B. Total Tonnage - If a pilot reports a fraction of tonnage, round the tonnage according to the previous rules set down (Example IV). If the tonnage reported is less than 1/2 ton total, the coder will code a positive response in the comments field (Example V).

EXAMPLE IV

Flight log example:

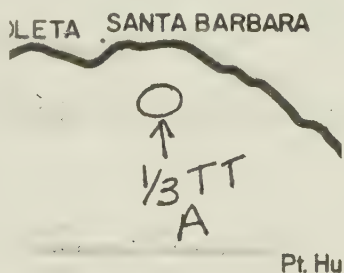


Coded example:

BLOCK NUMBER					CF & G SPECIES CODE					Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS					TOTAL TONNAGE										NUMBER OF INDIVIDUALS					SPECIES TABLE NO.
655					11012					12		2						4															1
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45		

EXAMPLE V

Flight log example:



Coded example:

BLOCK NUMBER					CF & G SPECIES CODE		Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE										NUMBER OF INDIVIDUALS					SPECIES TABLE NO.					
652					11011				6																	1					
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45

Usage: Sighting (continued)

Combined Species Codes

Pilots sometimes record a sighting with a combination of species and assign a total tonnage for the sighting. For these cases, special combined species codes have been produced. For a complete list, see page 59.

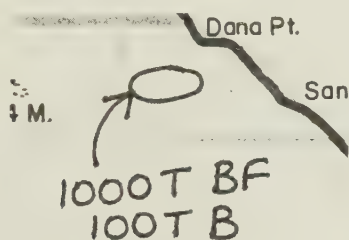
The combined species codes should be used only when the pilot makes no distinction between the tonnage or schools reported for each species.

Example I:

Pilot distinguishes between species

Use separate species code

Flight log example:



Dana Pt.

San

1000T BF

100T B

Coded example:

BLOCK NUMBER					CF & G SPECIES CODE		Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS					TOTAL TONNAGE										NUMBER OF INDIVIDUALS					SPECIES TABLE NO	
757					412		2		2						1000															1	
757					312		2		2						100															1	
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45

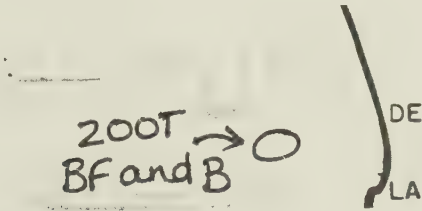
Usage: Sighting (continued)

Example II

Pilot does not distinguish between species

Use combined species code

Flight log example:



Coded example:

BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO
843	1134	12		2		200		1
14 15 16 17 18 19	20 21 22 23	24	25	26 27	28 29 30 31	32 33 34 35 36 37 38 39 40	41 42 43 44	45

Usage: Sighting (continued)

Mammal Sightings

Whale, porpoise, and seal sightings will be coded. Pilots do not estimate tonnage for these mammals, but they do estimate number of schools as well as number of individuals. Mammal sightings will be handled in one of two ways:

1. School estimate

If the pilot estimates the number of schools of whales or porpoise, it will be coded as number of schools.

2. Number estimate

If the pilot estimates the number of individuals in an area, it will be coded as number of individuals.

Examples:

A) 6 gray whales in Block # 843

B) 5 schools porpoise in Block # 702

C) 3 seals in Block # 707

D) 2 pods of whales in Block # 844

Coded Example:

BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO
843	2001	12		3			6	1
702	2050	12		1	5			1
707	3000	12		3			3	1
844	2000	12		1	2			1
14 15 16 17 18 19	20 21 22 23	24	25	26 27	28 29 30 31	32 33 34 35 36 37 38 39 40	41 42 43 44	45

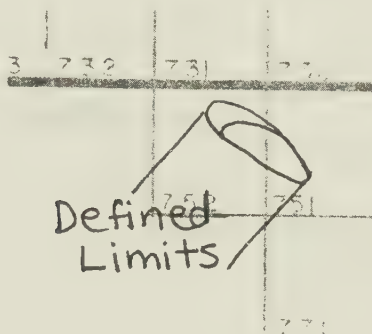
Usage: Sighting (continued)

Spots or Bunches

Some pilots refer to "spots" or "bunches" of fish. When this occurs, the coder will interpret a "spot" or "bunch" to be one school of fish.

Ambiguously Drawn Circles

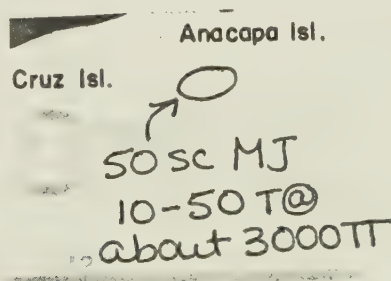
At times, a pilot will draw a circle on the flight log, depicting one school or a conglomerate of schools, that is unclear in nature as to its boundaries. When this happens, the boundaries of the sighting will be defined by the largest drawn circle.



Ambiguous Sighting Reports

Pilots normally report only one type of estimate for each sighting. They may, however, report more than one type of sighting. When this happens, the following definitions will apply.

Flight Log Example:



Usage: Sighting (continued)

Classify this example as a total tonnage sighting and code the total tonnage reported. Remember to include a positive response in the comments field.

Coded Example:

BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO
707	5511			2		3000		1
14 15 16 17 18 19	20 21 22 23	24	25	26 27	28 29 30 31	32 33 34 35 36 37 38 39 40	41 42 43 44	45

Flight Log Example:

GAVIOTA CAPITAN
GOLETA SANTA BAI
70
5sc A 30T@
about 200 T total

Include a positive response in the comments, and code as total tonnage.

Coded Example:

BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO
655	11011			2		200		1
14 15 16 17 18 19	20 21 22 23	24	25	26 27	28 29 30 31	32 33 34 35 36 37 38 39 40	41 42 43 44	45

Usage: Sighting (continued)

Ambiguous Classification

A. If a pilot ambiguously reports "several schools of fish" of Species X, but does report a total or individual school tonnage, record the total tonnage, and record a positive response in the comments field.

Flight log example:

MAVIOTA CAPITAN
GOLET
several sc A
200 TT

Coded example:

BLOCK NUMBER					CF & G SPECIES CODE					Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS					TOTAL TONNAGE										NUMBER OF INDIVIDUALS					SPECIES TABLE NO
655					11011							2						200															1
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45		

B. If a pilot reports "several schools of fish" of species X and does not report a tonnage, record a positive response in the comments field.

Flight log example:

many sc MB
UT
Pt. Huener

Coded example:

BLOCK NUMBER					CF & G SPECIES CODE					Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS					TOTAL TONNAGE										NUMBER OF INDIVIDUALS					SPECIES TABLE NO.
665					5111							6																					1
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45		

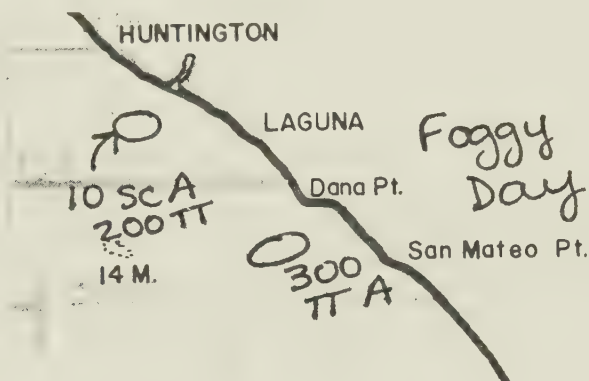
Usage: Sighting (continued)

Comment Code

The comment code is used to record any written information, not normally coded, that the pilot reports on the flight log. Some comments will be general in nature and pertain to the entire flight (Example A). Some written information is specific to a particular sighting and does not relate to the entire flight (Example B). When this is the case, the coder will record a positive response in the comments field for that sighting only.

Example A:

Flight Log Example:



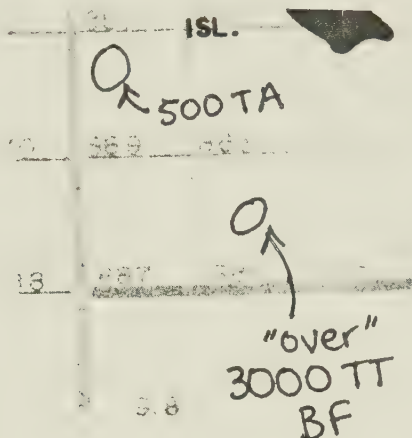
Coded Example:

BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO.
738	11012			4	10	200		1
757	11012			2		300		1
14 15 16 17 18 19	20 21 22 23	24	25	26 27	28 29 30 31	32 33 34 35 36 37 38 39 40	41 42 43 44	45

Usage: Sighting (continued)

Example B:

Flight Log Example:



Coded Example:

BLOCK NUMBER					CF & G SPECIES CODE		Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE					NUMBER OF INDIVIDUALS					SPECIES TABLE NO.										
886					411				2		3000										1										
869					11012				2		500										1										
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45

Comment Code - Expanded

The comment code in Column 23 helps to record the pilot's descriptions of the schools and tonnages sighted. Listed below are the codes and the specific comments to which they refer.

<u>Code</u>	<u>Comments</u>
1	Yes
2	No
3	Scattered
4	Many
5	Few
6	Heavy
7	Small
8	Jumpers
9	Some

Usage: Sighting (continued)

Red Tide

A species code number was established for the red tide and normal coding procedures were followed when the data coding began in February 1977. Later in the year, coding of the red tide was abolished and a positive comment code was indicated whenever the red tide was sighted. The reason for this action was due to the fact that some pilots indicated immense areas covered by the red tide on their flight logs. A positive comment code suffices in lieu of the many records necessary to retain this information. At one time, future plans for the AMRMS included a Comment Data Base from which specifics such as the red tide could be accessed.

Sonar-Spotted School

Occasionally, sightings will be recorded as "sonar-spotted" schools. This classification means that sometimes a pilot cannot see certain schools of fish or particular aspects of a school, but he may know they are there because a boat has spotted them using their sonar equipment. The pilot may then record the fish in the usual manner labeling the sighting as sonar-spotted. Do not code these particular sightings because they are not aerially-sighted. They should be coded as a positive comment in the effort record.

SIGHTING TYPE CODES
FOR THE
AERIAL MARINE RESOURCE MONITORING SYSTEM

<u>Code</u>	<u>Type</u>	<u>Explanation</u>
01	Type 1	Pilot recorded an estimated number of schools of one species of fish in one block.
02	Type 2	Pilot recorded a total tonnage for one species of fish in one block.
03	Type 3	Pilot recorded an estimated number of individuals of one species of fish in one block.
04	Type 4	Pilot recorded an estimated number of schools and a total tonnage for the entire number of schools of one species of fish in one block.
05	Type 5	Pilot recorded an estimated number of schools of one species of fish and the individual tonnage for <u>each</u> school in one block.
06	Type 6	Pilot only recorded a species name for one species of fish in one block.

PILOT REFERENCE TABLE
FOR THE
AERIAL MARINE RESOURCE MONITORING SYSTEM

<u>Numeric Code</u>	<u>Spotters</u>	<u>Alpha Code</u>
90	Unknown	---
1	Ed Durden	(ED)
2	John Bourgois	(JB)
3	Jack Mardesich	(JM)
4	Jack Whalen	(JW)
5	Leon Durden	(LD)
6	John O'Connor	(OC)
7	Paul Mardesich	(PM)
8	Tony Marinkovich	(TM)
9	Joe Miles	(JOM)
10	Roger Hillhouse	(RH)
11	Chuck Rogers	(CR)
12	C. L. Devillier	(CD)
13	Jim Squire	(JS)
14	Jim Ferreri	(JF)
15	Andy Witmer	(AW)
16	Frank Iacono	(FI)
17	Tom Greene	(TG)
18	Tom Wilson	(TW)
19	Mike Nix	(MN)

PLANE IDENTIFICATION
FOR THE
AERIAL MARINE RESOURCE MONITORING SYSTEM

<u>Numeric Code</u>	<u>Plane Number or Name</u>
001	"Flying Cloud"
002	"Sockeye"
003	"Bimbo"
004	1282
005	1832
006	73666
007	86489
008	1283 A
009	128 B
010	2196 B
011	7161 B
012	1677 C
013	9330 C
014	2448 D
015	8069 D
016	8589 D
017	4879 E
018	7547 F
019	1668 G
020	90216 G
021	1889 P or "Champion"

022	2869 Q
023	356 X
024	5236 X
025	2737 Y
026	1621 H
027	2454 J
028	XBHUD
029	7847 C
030	2827 A
031	1883 A
032	9987 R
033	XB-ZEP
034	FIN
035	4306 R
036	4705 K
037	5847 G
038	8741 V
039	9445 B
040	45 B
041	6293 R
042	1363 F
043	5020 H
044	5193 K
045	6543 V
046	60454
047	4789 R
048	76048

099

Unknown

CALIFORNIA FISH AND GAME
MARINE FISHERIES STATISTICS
SPECIES CODES
(Revised List)

Numerically Sequenced Species Code Numbers

<u>Numeric Code</u>	<u>Alpha Code</u>	<u>Identification</u>
001	YF	Yellowfin
002	SK	Skipjack Tuna
003	B	Bonito
004	BF	Bluefin Tuna
005	AL	Albacore Tuna
006	TU	Tuna (unclassified)
008	BI	Bigeye Tuna
019	BL	Bullet Mackerel
040	Y	Yellowtail
050	MU	Mackerel (unclassified)
051	MB or MP	Blue or Pacific Mackerel
055	MJ or MS	Jack or Spanish Mackerel
091	SF	Swordfish (Broadbill)
092	M	Marlin
095	SL	Sailfish
100	S	Sardine
110	A	Anchovy
130	BA	Barracuda
150	SH	Shark (unclassified)
156	BS	Basking Shark

167	BSH	Blue Shark
176	GWS	Great White Shark
180	SM	Smelt (unclassified)
184	SJ	Jack Smelt
292	MO	Mola or Sunfish
300	SN	Salmon
335	ST	Striped Bass
349	BLF	Blackfish (small whales)
400	SB	White Seabass
402	SBU	Seabass (unclassified)
420	CR	Croaker
446	N	Needlefish or Saury
550	PE	Perch (unclassified)
552	BP	Black Perch
711	SQ	Squid
800	C	Crab (unclassified)
814	SHR	Shrimp (unclassified)
960	J	Jellyfish
998	BT	Bait (feed)

Combination Species Code Numbers

<u>Numeric Code</u>	<u>Identification</u>
1111	Bluefin and Skipjack Tuna
1112	Bluefin and Bigeye Tuna
1113	Bluefin, Skipjack and Yellowfin
1114	Bluefin and Albacore
1115	Bluefin, Albacore and Yellowfin
1117	Bluefin, Albacore and Skipjack
1118	Jack Mackerel and Sardine
1119	Spanish or Jack Mackerel and Blue Mackerel
1120	Bonito and Anchovy
1121	Jack Mackerel and Squid
1122	Anchovy and Bluefin
1123	Shrimp and Squid
1124	Albacore and Tuna (unclassified)
1125	Blue Mackerel and Sardine
1126	Anchovy, Jack Mackerel, and Squid
1127	Anchovy and Squid
1128	Anchovy and Blue Mackerel
1129	Bluefin and Yellowfin
1130	Bonito and Blue Mackerel
1131	Mackerel (unclassified) and Sardine
1132	Barracuda and Bonito
1133	Anchovy and Sardine
1134	Bonito and Bluefin

- 1135 Anchovy, Blue Mackerel, and Sardine
- 1136 Blue Mackerel, Jack Mackerel, and Sardine
- 1137 Marlin, Blue Mackerel, and Jack Mackerel
- 1138 Marlin, Blue Mackerel, Jack Mackerel, and Sardine
- 1139 Marlin and Sardine
- 1141 Barracuda and Yellowtail
- 1142 Bullet Mackerel, Bluefin, and Skipjack
- 1144 Yellowfin and Skipjack

Mammal Species Code Numbers

<u>Numeric Code</u>	<u>Alpha Code</u>	<u>Identification</u>
2000	WU	Whale (unclassified)
2001	GW	Gray Whale
2002	SW	Small Whale
2003	PW	Pilot Whale
2004	KW	Killer Whale
2050	P	Porpoise
3000	SE	Seal
3001	SEL	Sea Lion

Species Code Numbers
(Miscellaneous)

<u>Numeric Code</u>	<u>Alpha Code</u>	<u>Identification</u>
1116	RT	Red Tide
9001	UNK	Unknown Species
9999	ME	Meatballs

SUPPLEMENTAL LIST OF
CALIFORNIA DEPARTMENT OF FISH AND GAME
MARINE FISHERIES STATISTICS
SPECIES CODES

Alphabetical Order

FISH

<u>Species</u>	<u>Code</u>
Albacore	005
American Shad	325
Animal Food	992
Arrowtooth Flounder	201
Atlantic Bluefin Tuna Obs.	010
Bank Rockfish	663
Barred Sand Bass	278
Barred Surfperch	551
Basking Shark	156
Bat Ray	171
Big Skate	176
Bigeye Tuna	008
Bigmouth Sole	202
Black Croaker	421
Black Rockfish	252
Black Skipjack	009
Black Surfperch	552
Black Tip Shark	149
Black & Yellow Rockfish	251
Blackfin Tuna	012
Blackfish	349
Blackgill Rockfish	667
Blacksmith	479
Blenny Eel	451
Blue Rockfish	665
Blue Shark	167
Bluefin Tuna	004
Bocaccio	253
Bonefish	477
Bonito Shark	151
Bronzespotted Rockfish	662
Broomtail Grouper	432
Brown Rockfish	267
Brown Smoothhound	154
Buck Shad	327
Bullet Mackerel	019
Butter Sole	208
C-0 Turbot	237

Cabazon	261
Calico Rockfish	671
Calico Surfperch	560
California Barracuda	130
California Corbina	426
California Grunion	181
California Halibut	222
California Killifish	491
California Lizardfish	473
California Moray	452
California Needlefish	476
California Sheephead	145
California Skate	177
Canary Rockfish	247
Carp	345
Catfish	320
Chilipepper	254
China Rockfish	258
Chum Salmon	301
Civache	401
Common Mola	292
Common Thresher	155
Copper Rockfish	655
Cow Shark	160
Cowcod	245
Croaker	420
Curlfin Turbot	235
Deep Bodied Anchovy	111
Diamond Turbot	236
Dolphinfish	481
Dover Sole	211
Dusky Shark	164
Dwarf Surfperch	561
Eel	450
English Sole	206
Eulachon	188
Fantail Sole	204
Flag Rockfish	657
Flounder	230
Flyingfish	445
Frog	920
Garibaldi	482
Giant Kelpfish	501
Giant Sea Bass	280
Gopher Rockfish	263
Grass Rockfish	652
Gray Smoothhound Shark	179
Greenblotched Rockfish	661
Greenspotted Rockfish	255
Greenstriped Rockfish	654
Grenadiers	198
Grouper	430
Halfmoon	478
Halibut	220

Halibut Roe	899
Hardhead	346
Hitch	348
Honeycomb Rockfish	660
Horn Shark	169
Hornyhead Turbot	238
Jack Crevalle	041
Jack Mackerel	055
Jacks	042
Jacksmelt	184
Juvenile Sardine	101
Kahawai	601
Kelp Bass	277
Kelp Greenling	290
Kelpfishes	510
King Salmon	302
Lamprey	453
Leopard Shark	153
Large Jack Mackerel	056
Lingcod	195
Longfin Sanddab	226
Longjaw Mudsucker	483
Longtail Tuna	011
Mackerel Unclassified	050
Mexican Pompano	081
Midshipman	485
Miscellaneous Fish	999
Miscellaneous Trawl Fish	997
New Zealand Mackerel	058
New Zealand Pilchard	102
Night Smelt	187
Northern Anchovy	110
Ocean Perch	271
Ocean Whitefish	490
Olive Rockfish	651
Opah	467
Opaleye	475
Oriental Tuna	007
Pacific Amberjack	043
Pacific Angel Shark	165
Pacific Bonito	003
Pacific Butterfish	080
Pacific Cod - True Cod	197
Pacific Electric Ray	172
Pacific Hake	495
Pacific Halibut	221
Pacific Herring	121
Pacific Mackerel	051
Pacific Sanddab	227
Pacific Sardine	100
Pacific Saury	446
Pacific Staghorn Sculpin	272
Pacific Tomcod	196
Petrale Sole	209

Pile Surfperch	559
Pink Rockfish	653
Pink Salmon	303
Pink Surfperch	563
Pinkrose Rockfish	664
Porpoise	916
Queenfish	440
Rainbow Surfperch	562
Rainbow Trout - Steelhead	316
Ratfish	166
Ray	170
Red Rockfish	265
Redtail Surfperch	553
Rex Sole	207
Robalos	414
Rock Bass	275
Rock Sole	203
Rock Wrasse	146
Rockfish	250
Roe Shad	326
Rosy Rockfish	268
Round Herring	105
Rubberlip Surfperch	558
Sablefish	190
Sailfish	095
Salema	484
Salmon	300
Salmon Shark	168
Sand Sole	205
Sanddab	225
Sargo	480
Sculpin	260
Senorita	144
Sevengill Shark	162
Shark	150
Shiner Surfperch	554
Shortfin Corvina	427
Shortspine Thornyhead	262
Shovelnose Guitarfish	174
Sierra	052
Silver Salmon	304
Sixgill Shark	161
Skate	175
Skipjack Tuna	002
Slender Sole	210
Smelt	180
Smooth Hammerhead	158
Snapper - Mexico	415
Sole	200
Soupfin Shark	159
Southern Anchovy	113
Speckled Rockfish	669
Speckled Sanddab	228
Spiny Dogfish	152

Splitnose Rockfish	270
Splittail	347
Spotfin Croaker	422
Spotted Cabrilla	431
Spotted Cusk-Eel	455
Spotted Sand Bass	276
Spotted Turbot	239
Square Tail Catfish	322
Squawfish	365
Squarespot Rockfish	666
Starry Flounder	231
Starry Rockfish	256
Stickleback	361
Stingray	173
Striped Bass	335
Striped Marlin	092
Striped Mullet	135
Striped Rockfish	257
Stripetail Rockfish	668
Sturgeon	470
Sucker	375
Surf Smelt	182
Surfperch	550
Swell Shark	163
Swordfish	091
Swordspine Rockfish	670
Tai - Mexico	465
Terrapin	931
Thornback Skate	178
Threadfin Shad	324
Tongue Sole	212
Top Smelt	186
Totuava Sea Bass	410
Treefish	658
Trigger Fish	291
Turbot	240
Turtle	930
Unclassified Tuna	006
Vermilion Rockfish	249
Wahoo	057
Walleye Surfperch	557
White Croaker	435
White Seabass	400
White Surfperch	556
Whitebait Smelt	185
Whitebelly Rockfish	246
Widow Rockfish	269
Wolf-Eel	454
Yelloweye Rockfish	248
Yellowfin Croaker	423
Yellowfin Tuna	001
Yellowtail	040
Yellowtail Rockfish	259
Zebraperch	602

MOLLUSK

<u>Species</u>	<u>Code</u>
Abalone	700
Black Abalone	701
California Jacknife Clam	728
Clam	720
Common Littleneck	721
Common Washington Clam	727
Eastern Oyster	741
European Flat Oyster	745
Flat Abalone	708
Freshwater Clam	733
Freshwater Snail	781
Gaper Clam	726
Giant Pacific Oyster	743
Green Abalone	703
Jumbo Squid	710
Kumamoto Oyster	744
Market Squid	711
Miscellaneous Mollusk	799
Mussel	730
Native Oyster	742
Northern Quahog	737
Octopus	712
Oyster	740
Pencil Clams	735
Pink Abalone	704
Pinto Abalone	707
Pismo Clam	722
Razor Clam	725
Red Abalone	702
Rock Scallop	718
Scallop	719
Sea Snail	732
Soft Shell Clam	723
Starfish	738
Threaded Abalone	706
White Abalone	705

ECHINODERM

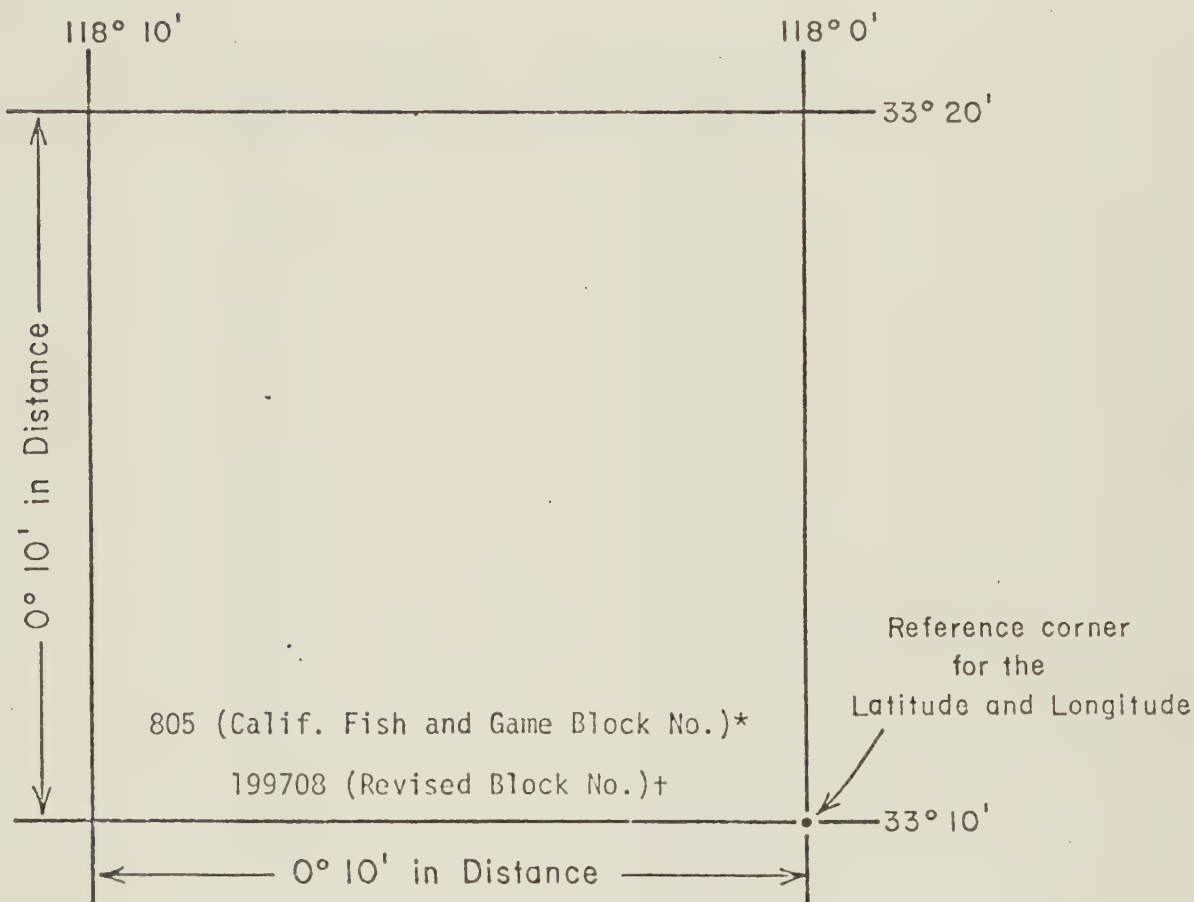
<u>Species</u>	<u>Code</u>
Echinoderm	750
Sea Star	751
Sea Urchin	752

CRUSTACEAN

<u>Species</u>	<u>Code</u>
Bay Shrimp	810
Brine Shrimp	819
California Spiny Lobster	820
Crayfish	825
Dungeness Crab	800
Ghost Shrimp	811
King Crab	804
Pacific Ocean Shrimp	812
Pelagic Red Crab	807
Ridgeback Prawn	813
Rock Crab	801
Shrimp Unclassified	814
Spider Crab	803
Spot Prawn	815
Tanner Crab	808

AERIAL MARINE RESOURCE MONITORING SYSTEM BLOCK NUMBERING METHOD

Example: 10' Sq. Block



Block No. Calculation Procedure

Latitude = $33^{\circ} 10'$

Longitude = 118°

- 1) change degrees to minutes
 $33 \times 60 = 1980'$
- 2) add any excess minutes
 $1980' + 10' = 1990'$
- 3) divide by 10
 $1990 \div 10 = 199$

End quotient is first three digits of block no.

- 1) change degrees to minutes
 $118 \times 60 = 7080$
- 2) add any excess minutes
 $7080 + 0 = 7080$
- 3) divide by 10
 $7080 \div 10 = 708$

End quotient is last three digits of block no.

* Developed in 1935

† Developed in 1977

CALIFORNIA FISH AND GAME
BLOCK NUMBERS
AND
THEIR CORRESPONDING
MACCALL-CARUSO BLOCK NUMBERS

California Fish & Game <u>Block</u> <u>Number</u> (<u>old</u>)	MacCall-Caruso <u>Block</u> <u>Number</u> (<u>new</u>)
472	224734
473	224735
474	224736
475	224737
476	224738
477	224739
478	223734
479	223735
480	223736
481	223737
482	223738
483	223739
501	222733
502	222734
503	222735
504	222736
505	222737
506	222738
507	221730
508	221731
509	221732
510	221733
511	221734
512	221735
513	221736
514	221737
515	221738
516	220730
517	220731
518	220732
519	220733
520	220734
521	220735
522	220736
523	220737
524	220738
525	219730
526	219731
527	219732
528	219733
529	219734
530	219735
531	219736
532	218731

533	218732
534	218733
535	218734
536	218735
537	218736
538	217730
539	217731
540	217732
541	217733
542	217734
543	217735
544	217736
547	216729
548	216730
549	216731
550	216732
551	216733
552	216734
553	215728
554	215729
555	215730
556	215731
557	215732
558	215733
559	215734
560	214727
561	214728
562	214729
563	214730
564	214731
565	214732
566	214733
567	214734
601	213726
602	213727
603	213728
604	213729
605	213730
606	213731
607	212725
608	212726
609	212727
610	212728
611	212729
612	212730
613	212731
614	211724
615	211725
616	211726
617	211727
618	211728
619	211729
620	211730
621	211731

622	210723
623	210724
624	210725
625	210726
626	210727
627	210728
628	210729
629	210730
630	210731
631	209723
632	209724
633	209725
634	209726
635	209727
636	209728
637	208723
638	208724
639	208725
640	208726
641	208727
642	208728
643	207723
644	207724
645	207725
646	207726
647	207727
648	207728
651	206716
652	206717
653	206718
654	206719
655	206720
656	206721
657	206722
658	206723
659	206724
660	206725
661	206726
662	206727
663	206728
664	205715
665	205716
666	205717
667	205718
668	205719
669	205720
670	205721
671	205722
672	205723
673	205724
674	205725
675	205726
676	205727
677	205728

678	204710
679	204711
680	204712
681	204713
682	204714
683	204715
684	204716
685	204717
686	204718
687	204719
688	204720
689	204721
690	204722
691	204723
692	204724
693	204725
694	204726
695	204727
696	204728
701	203710
702	203711
703	203712
704	203713
705	203714
706	203715
707	203716
708	203717
709	203718
710	203719
711	203720
712	203721
713	203722
714	203723
715	203724
716	203725
717	203726
718	202708
719	202709
720	202710
721	202711
722	202712
723	202713
724	202714
725	202715
726	202716
727	202717
728	202718
729	202719
730	202720
731	202721
732	202722
733	202723
734	202724
735	202725

736	202726
737	201706
738	201707
739	201708
740	201709
741	201710
742	201711
743	201712
744	201713
745	201714
746	201715
747	201716
748	201717
749	201718
750	201719
751	201720
752	201721
753	201722
754	201723
755	201724
756	200705
757	200706
758	200707
759	200708
760	200709
761	200710
762	200711
763	200712
764	200713
765	200714
766	200715
767	200716
768	200717
769	200718
770	200719
771	200720
772	200721
773	200722
774	200723
775	200724
801	199704
802	199705
803	199706
804	199707
805	199708
806	199709
807	199710
808	199711
809	199712
810	199713
811	199714
812	199715
813	199716
814	199717

815	199718
816	199719
817	199720
818	199721
819	199722
820	199723
821	198703
822	198704
823	198705
824	198706
825	198707
826	198708
827	198709
828	198710
829	198711
830	198712
831	198713
832	198714
833	198715
834	198716
835	198717
836	198718
837	198719
838	198720
839	198721
840	198722
841	198723
842	197703
843	197704
844	197705
845	197706
846	197707
847	197708
848	197709
849	197710
850	197711
851	197712
852	197713
853	197714
854	197715
855	197716
856	197717
857	197718
858	197719
859	196702
860	196703
861	196704
862	196705
863	196706
864	196707
865	196708
866	196709
867	196710
868	196711

869	196712
870	196713
871	196714
872	196715
873	196716
874	196717
875	196718
876	196719
877	195702
878	195703
879	195704
880	195705
881	195706
882	195707
883	195708
884	195709
885	195710
886	195711
887	195712
888	195713
889	195714
890	195715
891	195716
892	195717
893	195718
894	195719
901	194702
902	193702
903	193701
904	193703
916	194703

LIST OF ALL
POSSIBLE BLOCKS
IN ORDER
(MacCall-Caruso Block Numbers)

990
991
992
993
994
995
999
167689
167690
167691
167692
167693
167694
167695
167696
167697
167698
167699
167700
167701
167702
168689
168690
168691
168692
168693
168694
168695
168696
168697
168698
168699
168700
168701
168702
169689
169690
169691
169692
169693
169694
169695
169696
169697
169698
169699
169700

169701
169702
170689
170690
170691
170692
170693
170694
170695
170696
170697
170698
170699
170700
170701
170702
171689
171690
171691
171692
171693
171694
171695
171696
171697
171698
171699
171700
171701
171702
172689
172690
172691
172692
172693
172694
172695
172696
172697
172698
172699
172700
172701
172702
173689
173690
173691
173692
173693
173694
173695
173696
173697
173698

173699
173700
173701
173702
174689
174690
174691
174692
174693
174694
174695
174696
174697
174698
174699
174700
174701
174702
175689
175690
175691
175692
175693
175694
175695
175696
175697
175698
175699
175700
175701
175702
176689
176690
176691
176692
176693
176694
176695
176696
176697
176698
176699
176700
176701
176702
177691
177692
177693
177694
177695
177696
177697
177698

177699
177700
177701
177702
178693
178694
178695
178696
178697
178698
178699
178700
178701
178702
178703
178704
178705
178706
178707
178708
178709
178710
178711
179694
179695
179696
179697
179698
179699
179700
179701
179702
179703
179704
179705
179706
179707
179708
179709
179710
179711
180694
180695
180696
180697
180698
180699
180700
180701
180702
180703
180704
180705
180706

180707
180708
180709
180710
180711
181694
181695
181696
181697
181698
181699
181700
181701
181702
181703
181704
181705
181706
181707
181708
181709
181710
181711
182695
182696
182697
182698
182699
182700
182701
182702
182703
182704
182705
182706
182707
182708
182709
182710
182711
183696
183697
183698
183699
183700
183701
183702
183703
183704
183705
183706
183707
183708
183709

183710
183711
183712
184696
184697
184698
184699
184700
184701
184702
184703
184704
184705
184706
184707
184708
184709
184710
184711
184712
185697
185698
185699
185700
185701
185702
185703
185704
185705
185706
185707
185708
185709
185710
185711
185712
186698
186699
186700
186701
186702
186703
186704
186705
186706
186707
186708
186709
186710
186711
186712
187698
187699
187700

187701
187702
187703
187704
187705
187706
187707
187708
187709
187710
187711
187712
188698
188699
188700
188701
188702
188703
188704
188705
188706
188707
188708
188709
188710
188711
188712
189700
189701
189702
189703
189704
189705
189706
189707
189708
189709
189710
189711
189712
189713
190699
190700
190701
190702
190703
190704
190705
190706
190707
190708
190709
190710
190711

190712
190713
190714
191699
191700
191701
191702
191703
191704
191705
191706
191707
191708
191709
191710
191711
191712
191713
191714
192701
192702
192703
192704
192705
192706
192707
192708
192709
192710
192711
192712
192713
192714
192715
192716
193701
193702
193703
193704
193705
193706
193707
193708
193709
193710
193711
193712
193713
193714
193715
193716
194702
194703
194704

194705
194706
194707
194708
194709
194710
194711
194712
194713
194714
194715
194716
195702
195703
195704
195705
195706
195707
195708
195709
195710
195711
195712
195713
195714
195715
195716
195717
195718
195719
196702
196703
196704
196705
196706
196707
196708
196709
196710
196711
196712
196713
196714
196715
196716
196717
196718
196719
197703
197704
197705
197706
197707
197708

197709
197710
197711
197712
197713
197714
197715
197716
197717
197718
197719
198703
198704
198705
198706
198707
198708
198709
198710
198711
198712
198713
198714
198715
198716
198717
198718
198719
198720
198721
198722
198723
199704
199705
199706
199707
199708
199709
199710
199711
199712
199713
199714
199715
199716
199717
199718
199719
199720
199721
199722
199723
200705
200706

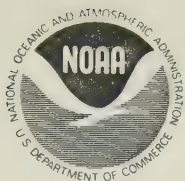
200707
200708
200709
200710
200711
200712
200713
200714
200715
200716
200717
200718
200719
200720
200721
200722
200723
200724
201706
201707
201708
201709
201710
201711
201712
201713
201714
201715
201716
201717
201718
201719
201720
201721
201722
201723
201724
202708
202709
202710
202711
202712
202713
202714
202715
202716
202717
202718
202719
202720
202721
202722
202723
202724

202725
202726
203710
203711
203712
203713
203714
203715
203716
203717
203718
203719
203720
203721
203722
203723
203724
203725
203726
204710
204711
204712
204713
204714
204715
204716
204717
204718
204719
204720
204721
204722
204723
204724
204725
204726
204727
204728
205715
205716
205717
205718
205719
205720
205721
205722
205723
205724
205725
205726
205727
205728
206716
206717

206718
206719
206720
206721
206722
206723
206724
206725
206726
206727
206728
207723
207724
207725
207726
207727
207728
208723
208724
208725
208726
208727
208728
209723
209724
209725
209726
209727
209728
210723
210724
210725
210726
210727
210728
210729
210730
210731
211724
211725
211726
211727
211728
211729
211730
211731
212725
212726
212727
212728
212729
212730
212731
213726

213727
213728
213729
213730
213731
214727
214728
214729
214730
214731
214732
214733
214734
215728
215729
215730
215731
215732
215733
215734
216729
216730
216731
216732
216733
216734
217730
217731
217732
217733
217734
217735
217736
218731
218732
218733
218734
218735
218736
219730
219731
219732
219733
219734
219735
219736
220730
220731
220732
220733
220734
220735
220736
220737

220738
221730
221731
221732
221733
221734
221735
221736
221737
221738
222733
222734
222735
222736
222737
222738
223734
223735
223736
223737
223738
223739
224734
224735
224736
224737
224738
224739

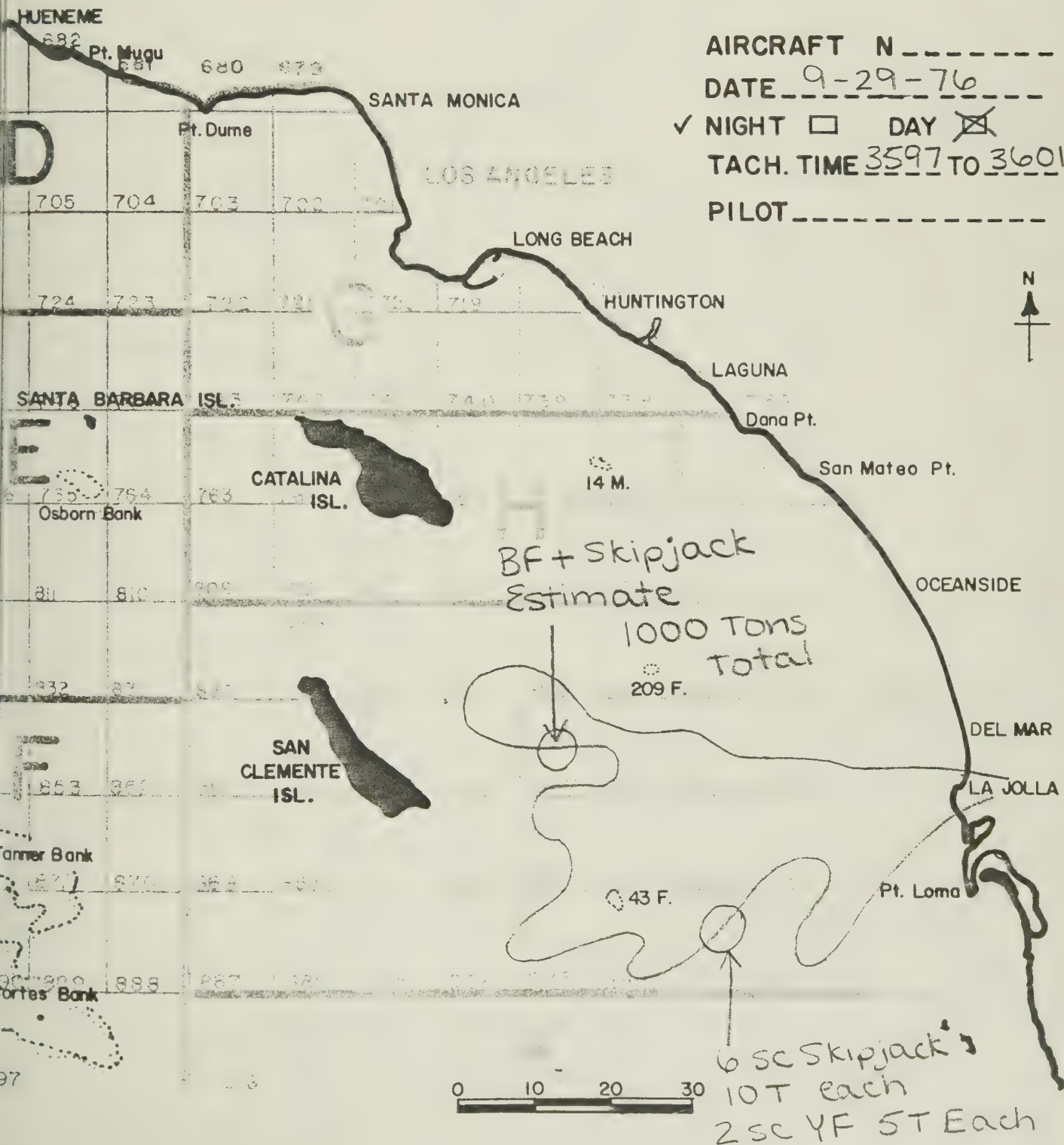


U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration

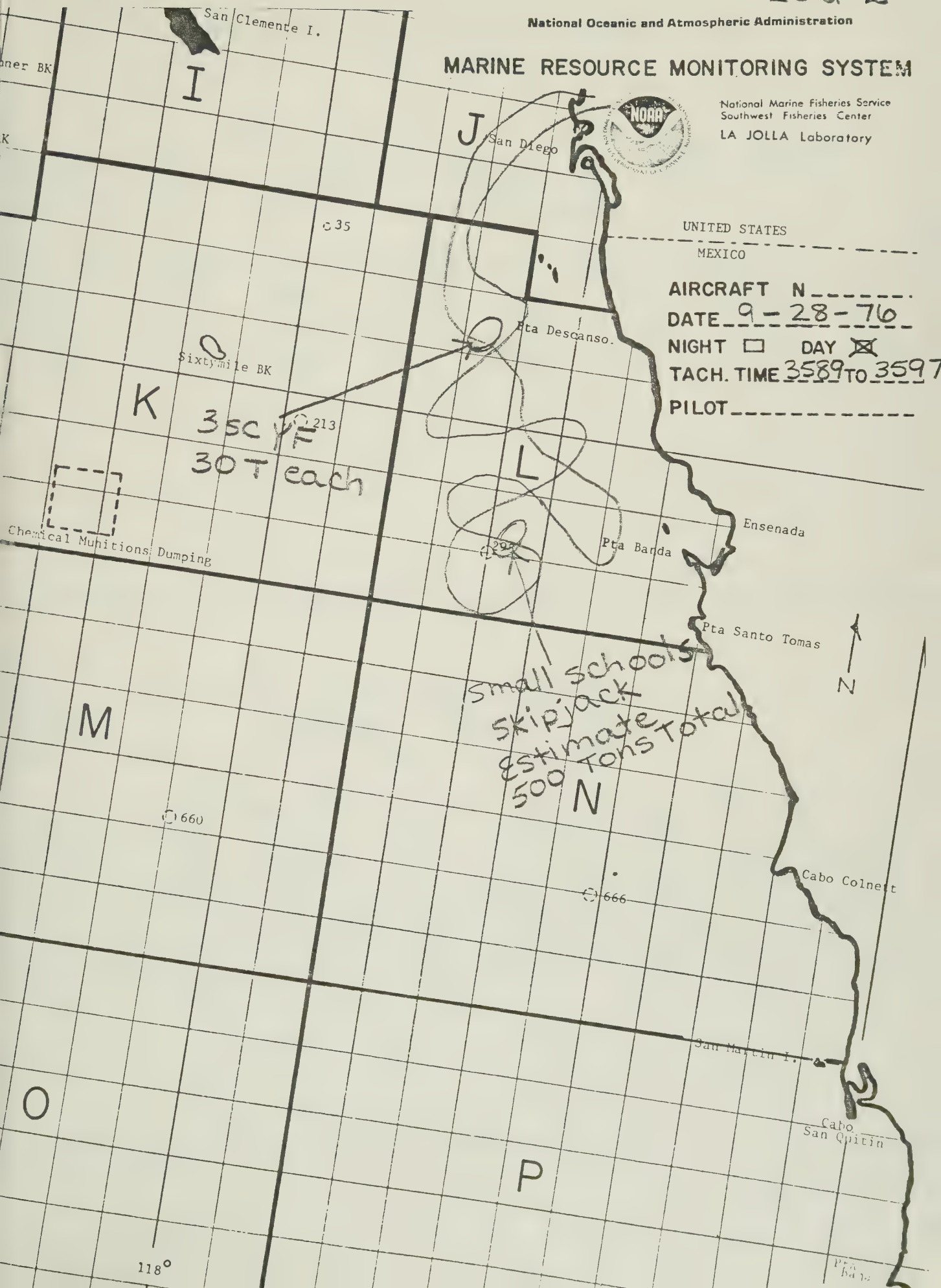
MARINE RESOURCE MONITORING SYSTEM

National Marine Fisheries Service
Southwest Fisheries Center
LA JOLLA Laboratory

TURA



MARINE RESOURCE MONITORING SYSTEM



DATE		PILOT				Flight
MONTH	DAY					
09	29					1
3 4	5 6	7	8	9	10	11

[illegible]

Date: 3/30/77

[illegible]

DATE		PILOT				Flight
MONTH	DAY					
29	28					
3	4	5	6	7	8	9 10 11

Day/Night	HOUR METER READING																										PLANE	Comments
	TIME OUT													TIME IN														
				3	5	3	9	0					3	5	9	7	0									2		
	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30									

[illegible]

Coder: Caruso
Date: 3/30/77

G 1

DATE				PILOT				Flight
MONTH		DAY						
0	9	2	9					1
3	4	5	6	7	8	9	10	11

[illegible]

G Z

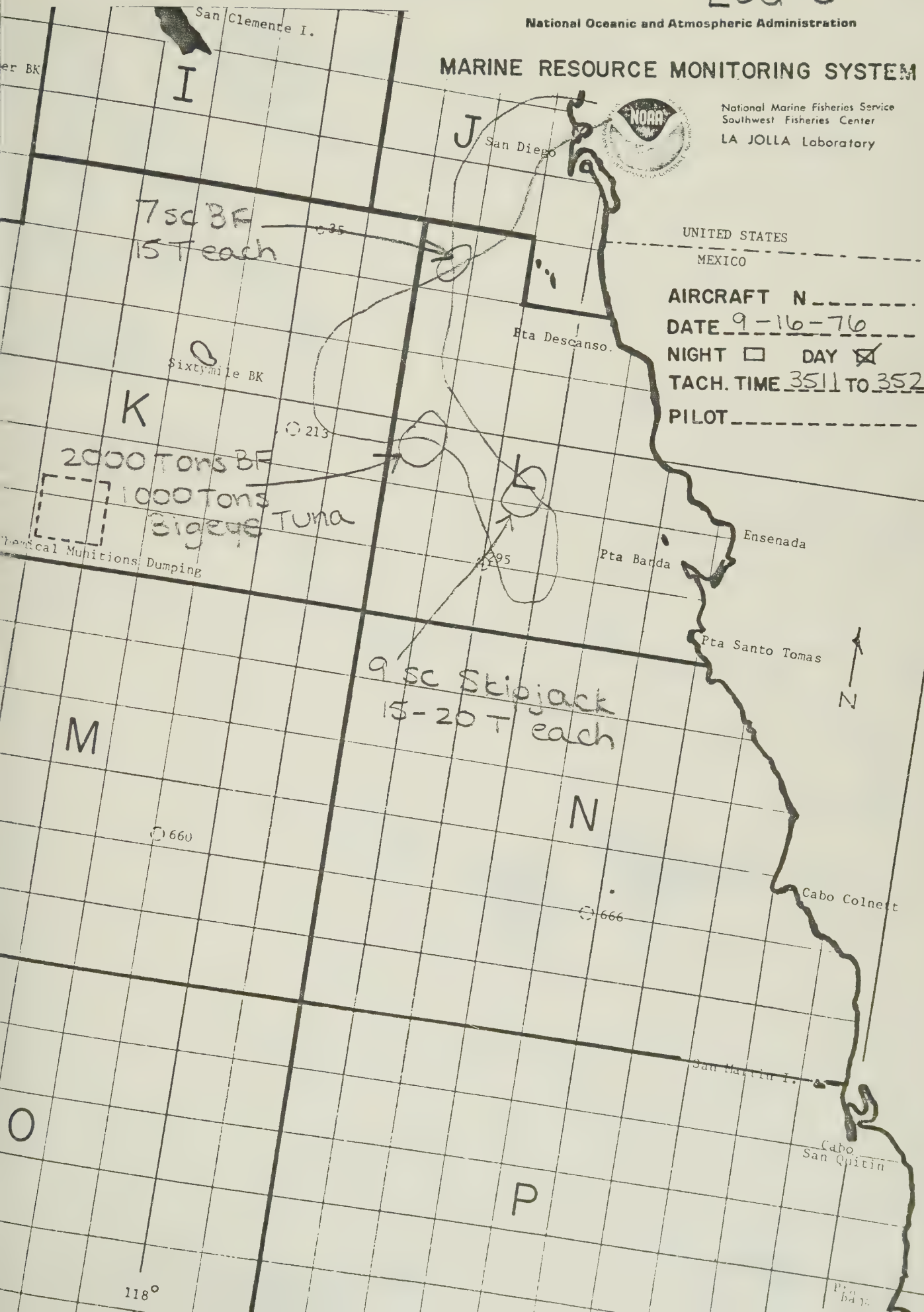
DATE				PILOT				Flight
MONTH		DAY						
0	9	2	8					;
3	4	5	6	7	8	9	10	11

[illegible]

National Oceanic and Atmospheric Administration

MARINE RESOURCE MONITORING SYSTEM

National Marine Fisheries Service
Southwest Fisheries Center
LA JOLLA Laboratory



UNITED STATES
MEXICO

AIRCRAFT N _____
DATE 9-16-76
NIGHT ☐ DAY ☒
TACH. TIME 3511 TO 3523
PILOT _____

Pt. Piedras Blancas



U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration

MARINE RESOURCE MONITORING SYSTEM

National Marine Fisheries Service
Southwest Fisheries Center

LA JOLLA Laboratory

Sheet 1

1 sc. An. 3TT
609 608 607
Morro Bay

Pt. Buchon

9 sc. MS 10-15 Tea.

AVILA

30 sc. An. 7-8 Tea.

45 sc. MS 10-15 Tea.

10 sc. An.

50 Tea.

624

635 634

12 sc MS
15 Tea.

640

639



AIRCRAFT N _____

DATE 7-24-65

✓ NIGHT ☐ DAY ☒

TACH. TIME 7667 TO 7673

PILOT _____

Pt. Sal

Pt. Arguello

GAVIOTA

CAPTAN

Pt. Concepcion

GOLETA

SANTA BARBARA

Scat. Bon.
est. 200 TT

Pt. Hueneme

San Miguel
Isl.

Santa Rosa Isl.

Santa Cruz Isl.

Anacapa Isl.

San
Nicolas
Isl.

0 10 20 30

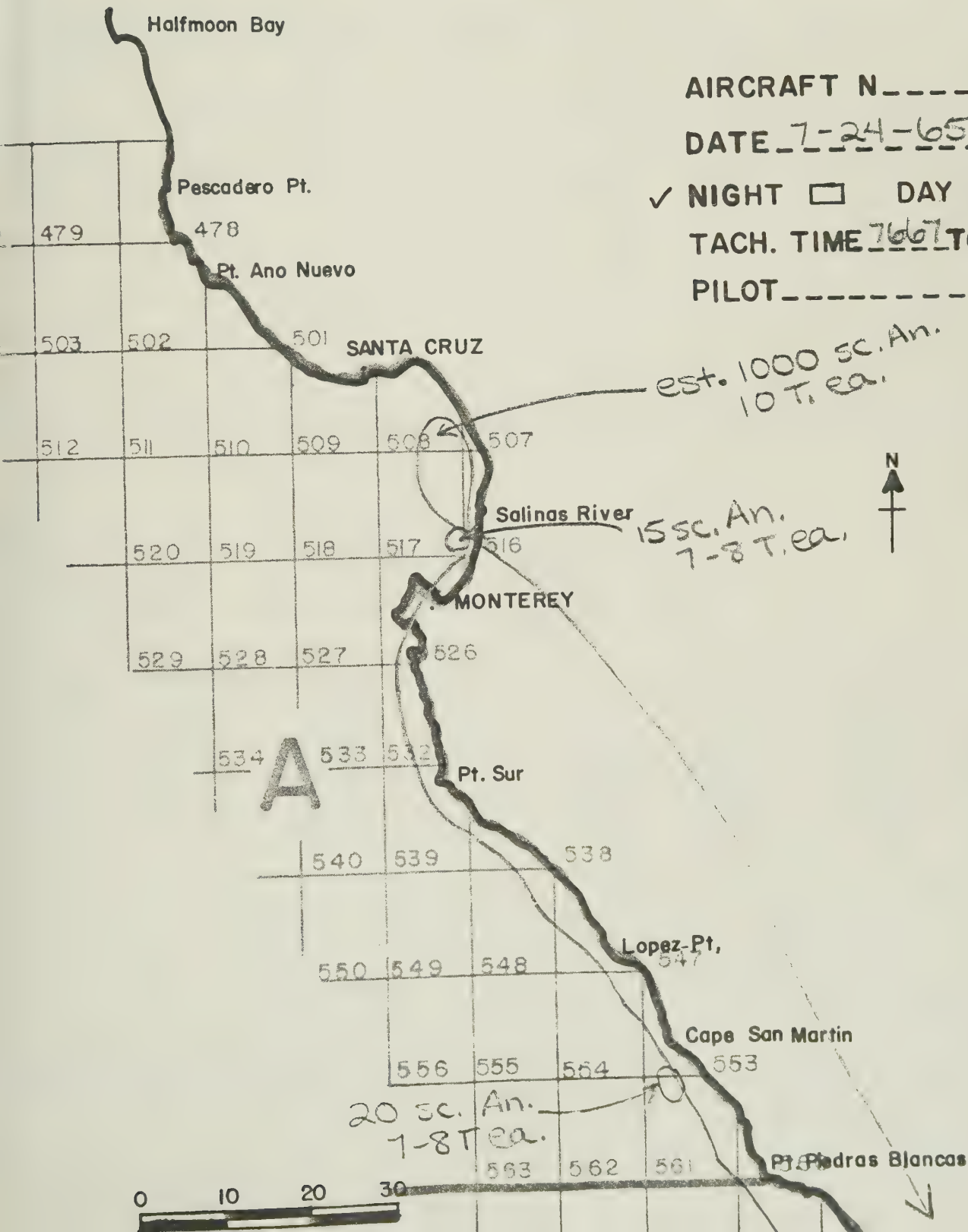


U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration

MARINE RESOURCE MONITORING SYSTEM

National Marine Fisheries Service
Southwest Fisheries Center
LA JOLLA Laboratory

Sheet 2



AIRCRAFT N-----

DATE 7-24-65

✓ NIGHT ☐ DAY ☒

TACH. TIME 7667 TO 7673

PILOT-----

Date: 3/30/77

LOG 3

DATE				PILOT				Flight
MONTH		DAY						
0	9	1	6					1
3	4	5	6	7	8	9	10	11

[illegible]

LOG 4

DATE				PILOT				Flight
MONTH		DAY						
0	7	2	4					
3	4	5	6	7	8	9	10	11

BLANKS (FILLERS)					BLOCK NUMBER	CF & G SPECIES CODE	Day/Night	Comments	SIGHTING TYPE	NUMBER OF SCHOOLS	TOTAL TONNAGE	NUMBER OF INDIVIDUALS	SPECIES TABLE NO
					655	3	13		2		150		1
					656	3	13		2		50		1
					622	55	12		5	45	563		1
					623	110	12		5	30	225		1
					623	55	12		5	9	113		1
					615	55	12		5	0	90		1
					607	55	12		5	6	90		1
					607	110	12		5	10	500		1
					607	110	12		5	0	300		1
					608	110	12		5	1	50		1

AERIAL MARINE RESOURCES MONITORING SYSTEM

SIGHTING RECORDS

Coder: Caruso
Date: 3/30/77

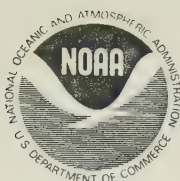
LOG 4 (cont.)

DATE						PILOT				Flight
R	MONTH		DAY							
5	0	7	2	4						
2	3	4	5	6	7	8	9	10	11	

[illegible]

DATE					PILOT	Flight
YR	MONTH	DAY				
2	3	4	5	6	7	8
					9	10
					11	

[illegible]

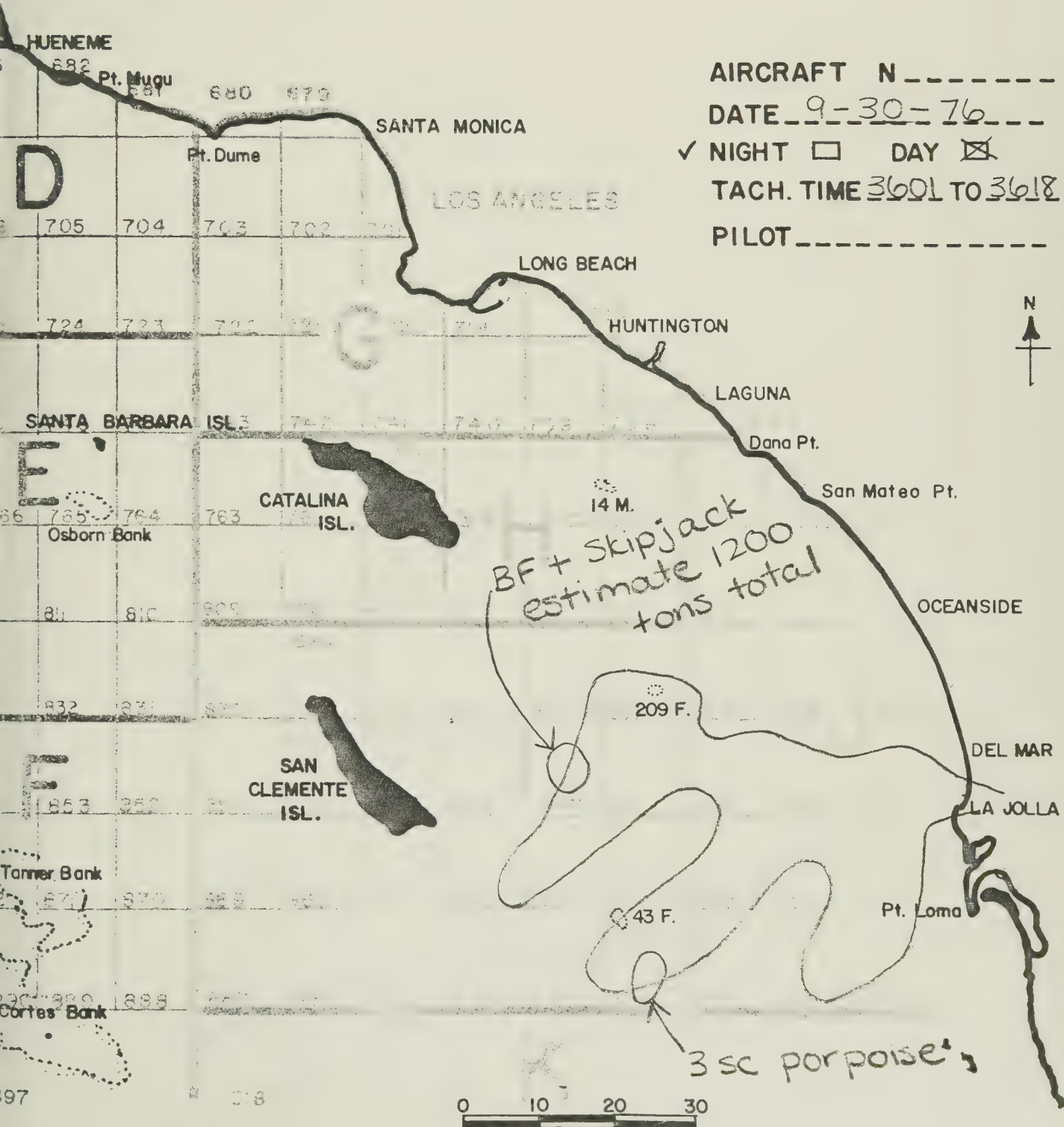


U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration

MARINE RESOURCE MONITORING SYSTEM

National Marine Fisheries Service
Southwest Fisheries Center
LA JOLLA Laboratory

NTURA



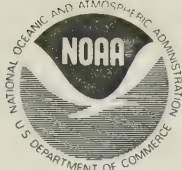
AIRCRAFT N _____

DATE 9-30-76

✓ NIGHT ☐ DAY ☒

TACH. TIME 3601 TO 3618

PILOT _____

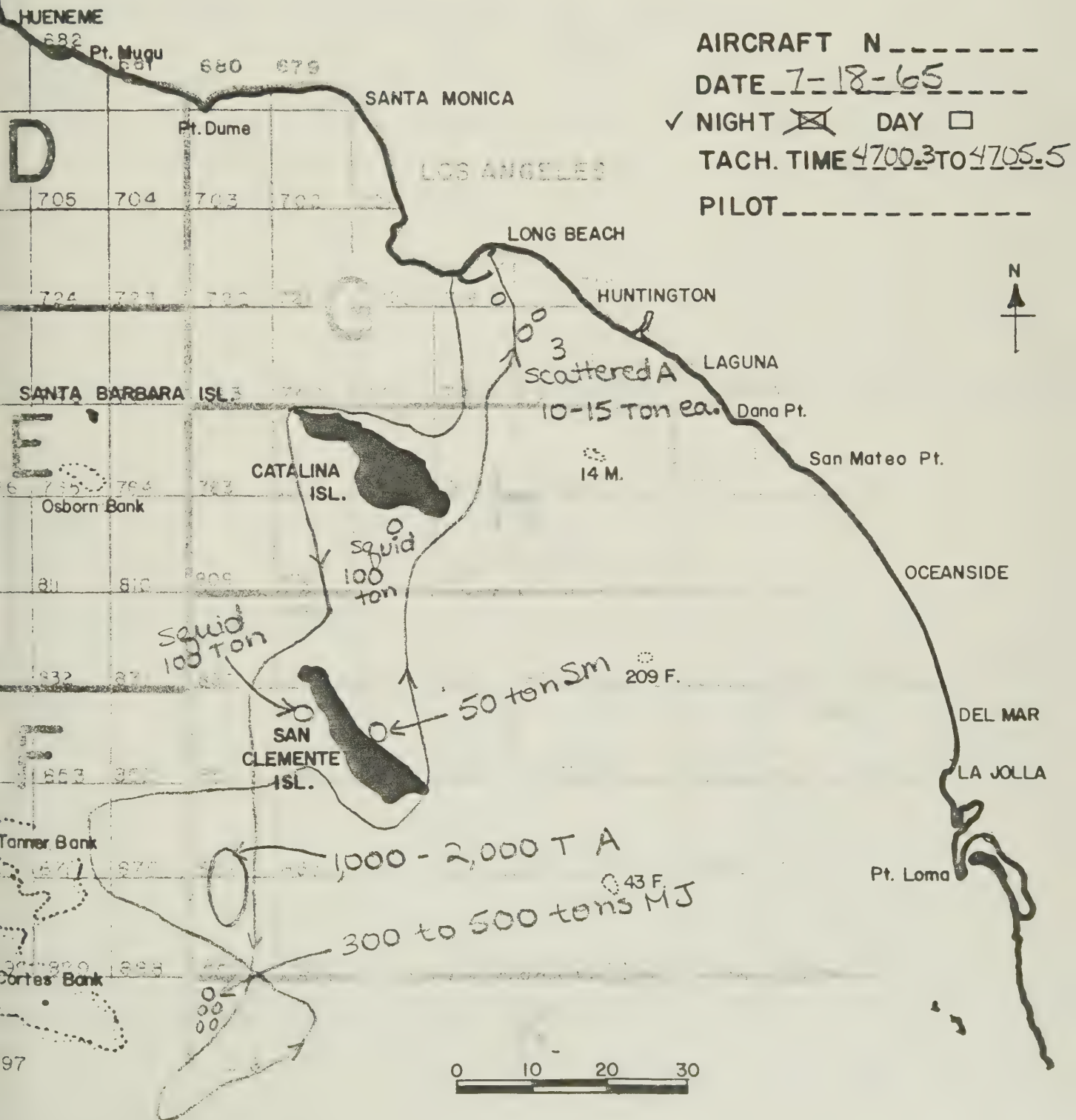


U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration

MARINE RESOURCE MONITORING SYSTEM

National Marine Fisheries Service
Southwest Fisheries Center
LA JOLLA Laboratory

ITURA



AERIAL MARINE RESOURCES MONITORING SYSTEM
EFFORT RECORDS

OG 5

DATE		PILOT				Flight
MONTH	DAY					
09	30					1
3	4	5	6	7	8	9 10 11

[illegible]

Coder: Caruso

Date: 3/30/77

[illegible]

OG 6

DATE		PILOT				Flight
MONTH	DAY					
07	18					1
3	4	5	6	7	8	9
				10		11

[illegible][illegible]

AERIAL MARINE RESOURCES MONITORING SYSTEM

SIGHTING RECORDS

Coder: Caruso
Date: 3/20/77

LOG 5

DATE				PILOT				Flight
MONTH		DAY						
0	9	3	0					
3	4	5	6	7	8	9	10	11

[illegible]

LOG 6

DATE				PILOT				Flight
MONTH		DAY						
07	18							
3	4	5	6	7	8	9	10	11

[illegible]

SUBDIRECTORIES

The AMRMS data resides on the VAX computer in account NM42. The data is divided into subgroups, each of which is stored in a separate subdirectory.

- [NM42] This is the main directory, where the data files are located while they are being edited.
Files: SIGHT**.DAT, EFFORT**.DAT, and miscellaneous files. (e.g. SIGHT83.DAT)
- [NM42.OLD] This directory contains all of the data files in their old (pre-1982) data formats (see Caruso, 1979). This data covers 1962 through part of 1982. The directory also contains the programs that were used to edit the data in the old formats.
- [NM42.EDIT] This directory contains all of the programs used in editing the data.
- [NM42.EFFORT] This directory contains all of the effort files from 1962 to the present. Only completely edited data is present. Files: E**.D (e.g. E62.D).
- [NM42.SIGHT] This directory contains all of the sighting files from 1962 to the present. Only completely edited data is present. Files: S**.D (e.g. S62.D).
- [NM42.FORMS] This directory contains the programs, form descriptions and form libraries necessary for entering AMRMS data using the Forms Management System of the VAX. The directory will also contain the data files created while entering data in this fashion.
- [NM42.DOC] This directory contains files used in preparing documentation which pertains to AMRMS data.

FILENAMES

When the data is first placed on the system, it will reside in files called EFFORT**.DAT and SIGHT**.DAT (e.g. EFFORT83.DAT, SIGHT83.DAT). As soon as the programs to translate block numbers (EDISK,SDISK) are run, the data file names become E**.D and S**.D (e.g. E83.D,S83.D). These are the final names of the files, and it indicates that the files are in the proper final form.

FMS

Form Management System produces a data form image on the terminal. Data can then be entered onto the form. Entering the AMRMS data in this fashion eliminates the need to fill out the coding forms by hand and the need to have these forms keypunched. The data entered using FMS goes directly into the computer. There is less chance for error this way, as only the coder has an opportunity to make an error. The traditional way of data entry, using coding sheets and keypunching, allows errors in both coding and keypunching.

Note: To use FMS, you must have a terminal that will act like a VT100.

name: EFFORT
form name:
t line: 1
line: 23
created: 8-MAR-83
r ID: 0
attributes: None
length: 452 bytes
er of fields: 14
re area size: 580 bytes

IMAGE

	1	2	3	4	5	6	7	8
1234567890123456789012345678901234567890123456789012345678901234567890								
11	EFFORT CODING SHEET							1
21								2
31	DATE:000000		PILOT:0000		FLIGHT:1			3
41								4
51	DAY/NIGHT CODE:		PLANE:000		COMMENTS:2			5
61								6
71	TIME OUT:000000.0		TIME IN:000000.0					7
81								8
91	BLOCKS							9
01	000000							10
11	000000							11
21	000000							12
31	000000							13
41	000000							14
51	000000							15
61								16
71								17
81								18
91								19
01								20
11								21
21								22
31								23
41								24
1234567890123456789012345678901234567890123456789012345678901234567890								
	1	2	3	4	5	6	7	8

VIDEO ATTRIBUTES

	1	2	3	4	5	6	7	8
1234567890123456789012345678901234567890123456789012345678901234567890								
11		22222222222222222222						1 1
21								1 2
31	22222000000		2222220000		22222220			1 3
41								1 4
51	2222222222222220		222222000		2222222220			1 5
61								1 6
71	22222222200000000		2222222200000000					1 7
81								1 8
91			222222					1 9
101			000000					110
111			000000					111
121			000000					112
131			000000					113
141			000000					114
151			000000					115
161								116
171								117
181								118
191								119
201								120
211								121
221								122
231								123
241								124
123456789012345678901234567890123456789012345678901234567890								
	1	2	3	4	5	6	7	8

Key to Video Attributes

Code	Attributes
0	Normal
2	Reverse video

Field Descriptions

- 3 11 Field D of length 6
Display attributes: Some Required, Autotab, Zero Fill
Field Type: Numeric
Clear character: '0'
Picture value: '999999'
- 3 35 Field PI of length 4
Display attributes: Some Required, Right Justified, Zero Fill
Field Type: Numeric
Clear character: '0'
Picture value: '9999'
- 3 57 Field F of length 1
Display attributes: Autotab
Field Type: Numeric
Clear character: ' '
Picture value: '9'
- 5 21 Field DN of length 1
Display attributes: Some Required, Autotab
Field Type: Numeric
Clear character: ' '
Picture value: '9'
- 5 35 Field PL of length 3
Display attributes: Some Required, Right Justified, Autotab, Zero Fill
Field Type: Numeric
Clear character: '0'
Picture value: '999'
- 5 59 Field C of length 1
Display attributes: Autotab
Field Type: Numeric
Clear character: ' '
Picture value: '9'
- 7 15 Field TD of length 7
Display attributes: Some Required, Right Justified, Zero Fill, Fixed Decimal
Field Type: Numeric
Clear character: '0'
Picture value: '999999.9'
- 7 37 Field TI of length 7
Display attributes: Some Required, Right Justified, Zero Fill, Fixed Decimal
Field Type: Numeric
Clear character: '0'
Picture value: '999999.9'

Field Descriptions (continued)

10 29 Field BL of length 6
Display attributes: Right Justified, Autotab, Zero Fill
Field Type: Numeric, Scrolled and repeated 6 times
Clear character: '0'
Picture value: '999999'

Form name: SIGHT
Help form name:
First line: 1
Last line: 23
Date created: 7-MAR-83
Owner ID: 0
Form attributes: None
Form length: 636 bytes
Number of fields: 47
Inquire area size: 952 bytes

IMAGE

	1	2	3	4	5	6	7	8
1234567890123456789012345678901234567890123456789012345678901234567890								
11	SIGHTING CODING SHEET							1 1
21								1 2
31	DATE:	PILOT:	FLIGHT:	DAY/NIGHT CODE:				1 3
41								1 4
51	SPECIES CODE TABLE:1							1 5
61								1 6
71	SPECIES		SIGHT	NUMBER OF		TOTAL	NUMBER OF	1 7
81	BLOCK	CODE	COMMENTS	TYPE	SCHOOLS	TONNAGE	INDIVIDUALS	1 8
91	000000	0000	2	00	0000	000000000	0000	1 9
101	000000	0000	2	00	0000	000000000	0000	110
111	000000	0000	2	00	0000	000000000	0000	111
121	000000	0000	2	00	0000	000000000	0000	112
131	000000	0000	2	00	0000	000000000	0000	113
141	000000	0000	2	00	0000	000000000	0000	114
151								115
161								116
171								117
181								118
191								119
201								120
211								121
221								122
231								123
241								124
1234567890123456789012345678901234567890123456789012345678901234567890								
	1	2	3	4	5	6	7	8

VIDEO ATTRIBUTES

	1	2	3	4	5	6	7	8	
1234567890123456789012345678901234567890123456789012345678901234567890									
11			22222222222222222222						1 1
21									1 2
31	22222000000	222220000		2222220		222222222222220			1 3
41									1 4
51		2222222222222222220							1 5
61									1 6
71		2222222		22222	222222222	222222222	2222222222		1 7
81	222222	2222222	22222222	22222	222222222	222222222	2222222222		1 8
91	000000	0000	0	00	0000	000000000	0000		1 9
01	000000	0000	0	00	0000	000000000	0000		1 10
11	000000	0000	0	00	0000	000000000	0000		1 11
21	000000	0000	0	00	0000	000000000	0000		1 12
31	000000	0000	0	00	0000	000000000	0000		1 13
41	000000	0000	0	00	0000	000000000	0000		1 14
51									1 15
61									1 16
71									1 17
81									1 18
91									1 19
01									1 20
11									1 21
21									1 22
31									1 23
41									1 24
1234567890123456789012345678901234567890123456789012345678901234567890									
	1	2	3	4	5	6	7	8	

Key to Video Attributes

Code	Attributes
0	Normal
2	Reverse video

Field Descriptions

- 3 11 Field D of length 6
Display attributes: Display Only, Autotab
Field Type: Numeric
Clear character: ' '
Picture value: '999999'
- 3 28 Field PI of length 4
Display attributes: Right Justified, Display Only, Autotab
Field Type: Numeric
Clear character: ' '
Picture value: '9999'
- 3 45 Field F of length 1
Display attributes: Display Only, Autotab
Field Type: Numeric
Clear character: ' '
Picture value: '9'
- 3 66 Field DN of length 1
Display attributes: Display Only, Autotab
Field Type: Numeric
Clear character: ' '
Picture value: '9'
- 5 41 Field SCT of length 1
Display attributes: Display Only, Autotab
Field Type: Numeric
Clear character: ' '
Picture value: '9'
- 9 3 Field B of length 6
Display attributes: Some Required, Right Justified, Zero Fill
Field Type: Numeric, Scrolled and repeated 6 times
Clear character: '0'
Picture value: '999999'
- 9 15 Field SC of length 4
Display attributes: Some Required, Right Justified, Zero Fill
Field Type: Numeric, Scrolled and repeated 6 times
Clear character: '0'
Picture value: '9999'
- 9 26 Field C of length 1
Display attributes: None
Field Type: Numeric, Scrolled and repeated 6 times
Clear character: ' '
Picture value: '9'

Field Descriptions (continued)

- 9 34 Field ST of length 2
Display attributes: Some Required, Right Justified, Zero Fill
Field Type: Numeric, Scrolled and repeated 6 times
Clear character: '0'
Picture value: '99'
- 9 46 Field NS of length 4
Display attributes: Right Justified, Zero Fill
Field Type: Numeric, Scrolled and repeated 6 times
Clear character: '0'
Picture value: '9999'
- 9 59 Field TT of length 9
Display attributes: Right Justified, Zero Fill
Field Type: Numeric, Scrolled and repeated 6 times
Clear character: '0'
Picture value: '999999999'
- 9 73 Field NI of length 4
Display attributes: Right Justified, Zero Fill
Field Type: Numeric, Scrolled and repeated 6 times
Clear character: '0'
Picture value: '9999'

CODE

AMRMS Coding Program

Purpose

The AMRMS coding program, CODE, allows the coding of pilot flight logs right on the computer. This eliminates the need for filling out the paper coding sheets and keypunching them.

Input

The CODE program is a data entry program. All you need is flight logs that have not yet been coded. When the program is run, the coding sheet is displayed on the screen. The data is entered the same as it is when filling out a paper coding sheet.

Each time CODE is run, two files are created; EFRM.DAT for effort data, SFRM.DAT for sighting data. Each of these files will have to be appended to the other sighting and effort files created by this program, and moved to [NM42] with filenames of EFFORT**.DAT and SIGHT**.DAT before the edit programs can be run.

To run CODE, a terminal that can mimic a VT100 is needed. Then get into the subdirectory [NM42.FORMS] by typing F. Next, type RUN CODE.


```
IMPLICIT INTEGER (A-Z)
DIMENSION IMPURE(32)
CHARACTER*1 F,DN,DE,SCT,CS(30),SIGHTINGS
CHARACTER*2 ST(30),FIELD
CHARACTER*3 PL
CHARACTER*4 PI,SC(30),NS(30),NI(30)
CHARACTER*6 D,BL(160),B(30)
CHARACTER*7 TO,TI
CHARACTER*9 TT(30),VALUE
OPEN(UNIT=10,NAME='EFORM.DAT',TYPE='NEW',FORM='FORMATTED',
RECORDTYPE='VARIABLE',RECL=993,CARRIAGECONTROL='LIST')
OPEN(UNIT=20,NAME='SFORM.DAT',TYPE='NEW',CARRIAGECONTROL='LIST')
9 CALL FDV$INIT(%DESCR(IMPURE),%REF(1000))
CHAN=2
CALL FDV$LCHAN(CHAN)
CALL FDV$LOPEN('AMRMS')
CALL FDV$CLRSH('EFFORT')
WI=1
I=1
N=1
DO 10 J=1,160
10 BL(J)='0000000'
11 CALL FDV$GET(D,TERM,'D')
IF (D.EQ.'999999') GO TO 999
12 CALL FDV$GET(PI,TERM,'PI')
IF (TERM.EQ.2) GO TO 11
13 CALL FDV$GET(F,TERM,'F')
IF (TERM.EQ.2) GO TO 12
14 CALL FDV$GET(DN,TERM,'DN')
IF (TERM.EQ.2) GO TO 13
15 CALL FDV$GET(PL,TERM,'PL')
IF (TERM.EQ.2) GO TO 14
16 CALL FDV$GET(DE,TERM,'C')
IF (TERM.EQ.2) GO TO 15
17 CALL FDV$GET(TO,TERM,'TO')
IF (TERM.EQ.2) GO TO 16
18 CALL FDV$GET(TI,TERM,'TI')
IF (TERM.EQ.2) GO TO 17
19 CALL FDV$GET(BL(I),TERM,'BL')
IF (TERM.EQ.4) GO TO 18
GO TO 21
20 IF (TERM.EQ.0) GO TO 23
CALL FDV$PFT(TERM,'BL')
GO TO 19
21 IF (TERM.LT.6) GO TO 20
IF ((TERM.EQ.7).OR.(TERM.EQ.9)) THEN
I=I-1
IF (I.EQ.0) THEN
I=1
CALL FDV$PUTL('TOP OF LIST')
GO TO 19
END IF
IF (I.GE.WI) GO TO 20
WI=WI-1
GO TO 22
```



```
END IF
IF ((TERM.EQ.6).OR.(TERM.EQ.8)) THEN
  I=I+1
  IF (I.GT.N) N=I
  IF (I.GT.160) THEN
    I=160
    CALL FDV$PUTL('END OF LIST')
    GO TO 19
  END IF
  IF (I.LE.(WI+5)) GO TO 20
  WI=WI+1
END IF
22 IF (BL(I).EQ.'000000') GO TO 20
CALL FDV$PFT(TERM,'BL')
CALL FDV$PUT(BL(I),'BL')
GO TO 19
23 CALL FDV$PUTL('IF CORRECT, HIT <CR>')
CALL FDV$GET(BL(I),TERM,'BL')
IF (TERM.NE.0) GO TO 19
24 WRITE(10,25)D,PI,F,DN,TO,TI,PL,CE,N,(BL(J),J=1,N)
25 FORMAT(A6,A4,2A1,2A7,A3,A1,I3,(N)A6)
CALL FDV$PUTL('WERE THERE ANY SIGHTINGS? (Y OR N) ')
READ(5,26)SIGHTINGS
26 FORMAT(A)
IF (SIGHTINGS.EQ.'N') GO TO 9
CALL FDV$INIT(%DESCR(IMPURE),%REF(1000))
CHAN=3
CALL FDV$LCHAN(CHAN)
CALL FDV$LOPEN('AMRMS')
CALL FDV$CLRSH('SIGHT')
WI=1
I=1
N=1
DO 27 J=1,30
  B(J)='000000'
  SC(J)='0000'
  CS(J)='2'
  ST(J)='00'
  NS(J)='0000'
  TT(J)='0000000000'
27 NI(J)='0000'
CALL FDV$PUT(D,'D')
CALL FDV$PUT(PI,'PI')
CALL FDV$PUT(F,'F')
CALL FDV$PUT(DN,'DN')
SCT='1'
FIELD='B'
28 CALL FDV$GET(VALUE,TERM,FIELD)
IF (FIELD.EQ.'B') B(I)=VALUE
IF (FIELD.EQ.'SC') SC(I)=VALUE
IF (FIELD.EQ.'C') CS(I)=VALUE
IF (FIELD.EQ.'ST') ST(I)=VALUE
IF (FIELD.EQ.'NS') NS(I)=VALUE
IF (FIELD.EQ.'TT') TT(I)=VALUE
IF (FIELD.EQ.'NI') NI(I)=VALUE
```



```
GO TO 30
29 IF (TERM.EQ.0) GO TO 32
CALL FDV$PFT(TERM, FIELD)
CALL FDV$GCF(FIELD)
GO TO 28
30 IF ((TERM.EQ.4).OR.(TERM.EQ.5)) THEN
CALL FDV$PUTL('INVALID TERMINATOR')
END IF
IF (TERM.LT.6) GO TO 29
IF ((TERM.EQ.7).OR.(TERM.EQ.9)) THEN
I=I-1
IF (I.EQ.0) THEN
I=1
CALL FDV$PUTL('TOP OF LIST')
GO TO 28
END IF
IF (I.GE.WI) GO TO 29
WI=WI-1
GO TO 31
END IF
IF ((TERM.EQ.6).OR.(TERM.EQ.8)) THEN
IF (((I+1).GT.N).AND.(TERM.EQ.8)) THEN
CALL FDV$PUTL('INVALID TERMINATOR')
GO TO 28
END IF
I=I+1
IF (I.GT.N) N=I
IF (I.GT.30) THEN
I=30
CALL FDV$PUTL('END OF LIST')
GO TO 28
END IF
IF (I.LE.(WI+5)) GO TO 29
WI=WI+1
END IF
31 IF ((FIELD.EQ.'B').AND.(B(I).EQ.'000000')) GO TO 28
IF ((FIELD.EQ.'SC').AND.(SC(I).EQ.'0000')) GO TO 28
IF ((FIELD.EQ.'ST').AND.(ST(I).EQ.'00')) GO TO 28
CALL FDV$PFT(TERM, FIELD)
CALL FDV$PUT(NI(I), 'NI')
CALL FDV$PUT(TT(I), 'TT')
CALL FDV$PUT(NS(I), 'NS')
CALL FDV$PUT(ST(I), 'ST')
CALL FDV$PUT(CS(I), 'C')
CALL FDV$PUT(SC(I), 'SC')
CALL FDV$PUT(B(I), 'B')
IF (TERM.EQ.6) FIELD='B'
GO TO 28
32 CALL FDV$PUTL('IF CORRECT, HIT (CR)')
CALL FDV$GET(VALUE, TERM, FIELD)
IF (TERM.NE.0) GO TO 28
33 DO 35 J=1, N
WRITE(20, 34) D, PI, F, B(J), SC(J), DN, CS(J), ST(J), NS(J),
1 TT(J), NI(J), SCT
34 FORMAT(A6, A4, A1, 2X, A6, A4, 2A1, A2, A4, A9, A4, A1)
```



```
35 CONTINUE  
GO TO 9  
959 CALL FDV$LCLOS  
STOP  
END
```


DATA EDITING

All data editing is done in [NM42].

A. SIGHTING FILES

The sighting files are punched in the same format that they will remain in.

1. SDISK

This program translates the block numbers, so the filename goes from SIGHT**.DAT (input) to S**.D (output). Errors may be fixed in the SOS or EDT editor on the VAX.

2. SORT (sight)

3. SEDIT

Any corrections can be made in the SOS or EDT editor on the VAX.

B. EFFORT FILES

The effort files may be punched in two ways, and this determines the order of data editing and programs to be used.

1. KEYPUNCHED

The keypuncher will be entering the data in fixed-length records. The records are in the format shown on the coding sheet.

a. SORT (fixed-length effort)

b. EDISK

This program translates the block numbers, so the filename goes from EFFORT**.DAT (input) to E**.D (output). It also changes the fixed-length records (several per flight) to variable-length records (one per flight). To fix any errors, first check the number of blocks in each flight. This can be done quickly using DATATRIEVE (DTR). If there are not any flights with more than 77 blocks, use the SOS editor on the VAX. If any flight has more than 77 blocks, DATATRIEVE must be used to make changes.

c. EEDIT

To make corrections, use the same method (either SOS editor or DTR) as above.

2. ENTERED ON THE FORMS MANAGEMENT SYSTEM

The data is entered into variable-length records.

a. FMSEDISK

This program translates the block numbers, so the filename goes from EFFORT**.DAT (input) to E**.D (output). To make corrections, use either the SOS editor or DTR, as explained above in 1b.

b. SORT (variable-length effort)

c. EEDIT

Make corrections using either the SOS editor or DTR, as explained above.

C. MERGE

This program does a cross check between the sighting and effort information. When finished editing these files, either append them to files already existing for the year in the EFFORT and SIGHT subdirectories, or simply move the files to those subdirectories if none exist for that year. These final files should be sorted.

To use any of the editing programs (EDISK, FMSEDISK, EEDIT, SDISK, SEDIT, MERGE):

- A. Get in subdirectory [NM42.EDIT] by typing E.
- B. Using an editor, substitute the desired input values into the program command file.
- C. Type @<program name>

SORT

Sort Command File

Purpose

SORT will sort any of the three file types used in AMRMS.

Input

The input may be of three types:

- A. Sighting file: fixed-length records. The file is sorted on the flight key, block number and species code.
- B. Effort file: fixed-length records. The file is sorted on the flight key and sequence number.
- C. Effort file: variable-length records. The file is sorted on the flight key.

The end result is a file that is sorted.

To use this command file, get in the subdirectory where the file to be sorted is, and type @[NM42.EDIT]SORT.

SORT.COM

PERFORMS SORT ON A CHOICE OF THE THREE TYPES OF AERIAL SURVEY
DATA FILES

GET CHOICE OF SORT TO BE PERFORMED
AND YEAR OF THE DATA FILE.

WRITE SYS\$OUTPUT " WHICH SORT? "
WRITE SYS\$OUTPUT " 1 SIGHTING FILE "
WRITE SYS\$OUTPUT " 2 EFFORT FILE WITH FIXED LENGTH RECORDS "
WRITE SYS\$OUTPUT " 3 EFFORT FILE WITH VARIABLE LENGTH RECORDS "
INQUIRE CHOICE " ENTER CHOICE "
INQUIRE YEAR " WHAT YEAR (2 DIGITS) "

BRANCH TO THE PROPER SORT

IF CHOICE .EQS. "1" THEN GOTO SIGHT
IF CHOICE .EQS. "2" THEN GOTO FIXEFFORT
IF CHOICE .EQS. "3" THEN GOTO VAREFFORT

***** SIGHT *****

SIGHT FILE IS SORTED ON:
FLIGHT KEY, BLOCK, SPECIES
FLIGHT KEY IS COMPOSED OF:
YEAR, MONTH, DAY, PILOT, FLIGHT

SIGHT:
FILE:="S"YEAR".D"
SORT/KEY=(POS=1,SIZE=23) 'FILE' 'FILE'
GOTO FINISH

***** FIXED LENGTH EFFORT *****

EFFORT FILE WITH FIXED LENGTH RECORDS IS SORTED ON:
FLIGHT KEY, CARD SEQUENCE
FLIGHT KEY IS COMPOSED OF:
YEAR, MONTH, DAY, PILOT, FLIGHT

FIXEFFORT:
FILE:="EFFORT"YEAR".DAT"
SORT/KEY=(POS=1,SIZE=11)/KEY=(POS=79,SIZE=2) 'FILE' 'FILE'
GOTO FINISH

***** VARIABLE LENGTH EFFORT *****

EFFORT FILE WITH VARIABLE LENGTH RECORDS IS SORTED ON:
FLIGHT KEY
FLIGHT KEY IS COMPOSED OF:

YEAR, MONTH, DAY, PILOT, FLIGHT

VAREFFORT:

FILE:="E" 'YEAR' ".D"

SORT/KEY=(POS=1,SIZE=11) 'FILE' 'FILE'

GOTO FINISH

***** FINISH *****

FINISH:

WRITE SYS\$OUTPUT " SORT COMPLETED "

SDISK

Sighting Data Program.

Purpose

The sighting program, SDISK, checks the block number recorded and changes any 3-digit code numbers to 6-digit numbers. Records are flagged if in error.

Input

SDISK uses four sets of input for its purposes. Set one is the raw data. Set two is a disk file with a list of block numbers originating from the California Department of Fish and Game. Next is a list of revised Fish and Game block numbers. The last set is a file of all possible 6-digit block numbers through which the pilots might fly. Sets 2-4 are sorted in ascending order. Since sets 2-4 must always be the same, the program automatically opens the files and reads the block numbers in.

As with the effort data, the end result of SDISK is a file of sighting data that can be used by the sighting verification program.

SDISK.COM

ET DEFAULT UDISK6:[NM42]

UN [NM42.EDIT]SDISK

GHT83.DAT

3.D

OST

XIT

PROGRAM: SDISK.FOR

THIS PROGRAM CREATES A DISK FILE OF THE AERIAL MARINE RESOURCES
MONITORING SYSTEM SIGHTING CARD IMAGES. IT WILL ALSO CHANGE ANY
THREE-DIGIT BLOCK NUMBERS TO SIX-DIGIT BLOCK NUMBERS BEFORE WRITING
THE CARD IMAGES ON DISK.

ORIGINAL PROGRAMMER: JOE CARUSO

REVISED FOR THE VAX SYSTEM BY SUSAN IACOMETTI IN APRIL 1982.

REVISED BY CINDY MEYER IN OCTOBER, 1982

```
COMMON/AA/ ONEWBK(778)
COMMON/BLOCKS/CFGBK(345), REVBK(345), BLOCK, Y, YEAR, MONTH, DAY, PILOT, N
CHARACTER*30 INFILE, OUTFILE
INTEGER CFGBK, REVBK, YEAR, DAY, PILOT, FLIGHT, BLOCK, SPECIE, DANITE,
*COMENT, STYPE, TOTAL, TABLE, Y, ONEWBK
```

GET FILENAMES

```
PRINT*, 'SIGHT FILE?'
READ(5, 88) INFILE
88 FORMAT(A)
PRINT*, 'OUTPUT SIGHT FILE?'
READ(5, 88) OUTFILE
```

OPEN ALL FILES

```
OPEN(UNIT=25, TYPE='OLD', NAME=INFILE, READONLY)
OPEN(UNIT=11, TYPE='NEW', NAME=OUTFILE, CARRIAGECONTROL='LIST')
OPEN(UNIT=9, TYPE='OLD', NAME=' [NM42.EDIT]BLOCK.DAT', READONLY)
OPEN(UNIT=15, TYPE='OLD', NAME=' [NM42.EDIT]ORB.DAT', READONLY)
```

READ THE OLD BLOCK NUMBERS FROM DISK
READ THE NEW BLOCK NUMBERS FROM DISK

```
READ(9, 1) (CFGBK(I), REVBK(I), I=1, 345)
```

READ ALL POSSIBLE SIX-DIGIT NUMBERS FROM DISK

```
READ(15, 2) (ONEWBK(I), I=1, 778)
```

INITIALIZE CARD COUNTER

```
N=0
```

INCREMENT THE CARD COUNTER

```
21 N=N+1
```

READ A SIGHTING CARD

```
READ(25, 3, END=99, ERR=98) YEAR, MONTH, DAY, PILOT, FLIGHT, BLOCK, SPECIE,
*DANITE, COMENT, STYPE, NOSC, TOTAL, IND, TABLE
```


TEST THE BLOCK NUMBER

CALL SEARCH

WRITE THE SIGHTING CARD IMAGE TO DISK

WRITE(11,3) YEAR,MONTH,DAY,PILOT,FLIGHT,BLOCK,SPECIE,DANITE,
*COMENT,STYPE,NOSC,TOTAL,IND,TABLE

BRANCH BACK AND READ ANOTHER CARD

GO TO 21

***** FORMAT STATEMENTS *****

1 FORMAT(2X,I3,2X,I6)

2 FORMAT(2X,I6)

3 FORMAT(3I2.2,I4.4,I1.1,2X,I6.6,I4.4,2I1.1,I2.2,I4.4,I9.9,I4.4,I1)

7 FORMAT(1H1,'***** SDISK HAS ENCOUNTERED ERRORS. AN ABNORMAL EXIT
* HAS BEEN MADE FROM THE PROGRAM *****')

8 FORMAT(1H1,'***** THE PROGRAM, SDISK, HAS EXECUTED SUCCESSFULLY
*****')

98 WRITE(6,7)

GO TO 100

99 WRITE(6,8)

100 CLOSE(UNIT=11)

STOP

END

SEARCH: A SUBROUTINE

SUBROUTINE SEARCH

THIS SUBROUTINE IS DESIGNED TO CHANGE THE THREE-DIGIT BLOCK NUMBER FROM THE AERIAL MARINE RESOURCE MONITORING SYSTEM TO THE SIX-DIGIT REVISED BLOCK NUMBER. THE SUBROUTINE ACCOMPLISHES THIS BY 1) MATCHING A THREE-DIGIT BLOCK NUMBER IN THE "CFGBK" ARRAY WITH THE BLOCK NUMBER INPUT FROM THE MAIN PROGRAM. AND 2) REPLACING THIS THREE-DIGIT NUMBER WITH A SIX-DIGIT NUMBER IN THE "REVBK" ARRAY. THE INDEX OF THE MATCHING NUMBER IN THE "CFGBK" ARRAY WILL INDEX THE CORRECT REPLACEMENT NUMBER IN THE "REVBK" ARRAY.

COMMON/AA/ ONEWBK(778)

COMMON/BLOCKS/CFGBK(345),REVBK(345),BLOCK,Y,YEAR,MONTH,DAY,PILOT,N
INTEGER X,Y,BLOCK,CFGBK,REVBK,HIGH,YEAR,DAY,PILOT,ONEWBK

SET THE BOTTOM OF THE SEARCH RANGE

LOW=0

IF(BLOCK.GT.989)GO TO 5

SET THE TOP OF THE SEARCH RANGE

HIGH=346

TEST FOR NO MATCH

2 IF((HIGH-LOW).LT.2)GO TO 3

SET THE MIDPOINT OF THE RANGE

$X=(HIGH-LOW)/2$

SET THE TEST INDEX

$Y=LOW+X$

TEST FOR IMMEDIATE MATCH

IF(BLOCK.EQ.CFGBK(Y))GO TO 4

RAISE THE BOTTOM OF THE RANGE

IF(BLOCK.GT.CFGBK(Y))LOW=Y

LOWER THE TOP OF THE RANGE

IF(BLOCK.LT.CFGBK(Y))HIGH=Y
GO TO 2

SET THE TOP OF THE SEARCH RANGE

5 HIGH=779

6 IF((HIGH-LOW).LT.2)GO TO 3

$X=(HIGH-LOW)/2$

$Y=LOW+X$

IF(BLOCK.EQ.ONEWBK(Y))GO TO 7

IF(BLOCK.GT.ONEWBK(Y))LOW=Y

IF(BLOCK.LT.ONEWBK(Y))HIGH=Y

GO TO 6

THE NUMBER IN QUESTION WAS NOT FOUND IN THE "CFGBK" ARRAY.
WRITE AN ERROR MESSAGE AND RETURN TO THE MAIN PROGRAM

3 WRITE(6,1)N,BLOCK, YEAR, MONTH, DAY, PILOT
RETURN

REPLACE THE OLD BLOCK NUMBER WITH THE REVISED BLOCK NUMBER

4 BLOCK=REVBK(Y)

7 RETURN

1 FORMAT(1H , 'CARD NO. = ', I5, ' HAS A BLOCK NO. = ', I6, ' THAT IS NOT
*IN THE MASTER LIST', / ' YEAR = ', I2, ' MONTH = ', I2, ' DAY = ',
*I2, ' PILOT = ', I4)
END

SEDIT

Sighting Verification Program

Purpose

The sighting verification program, SEDIT, checks the sighting data records for possible errors. Error messages tell the user on which record the possible error occurred and what variable was involved.

Input

Data input for SEDIT consists of three sets of data. The first is the raw sighting data file. The last two input variables are the year and month, respectively.

Verification Procedures

The data elements are tested and verified. This section describes the procedure involved.

- | | |
|-----------------|---|
| A. Year | <u>Input Data</u> - 2 digits punched in columns 8 and 9. From the first record read, the year is tested to make sure the correct file was accessed. Any difference causes program termination. Succeeding test failures cause error messages to be printed. |
| B. Month | <u>Input Data</u> - 2 digits punched in columns 8 and 9. Month is checked to see if the correct file is accessed. Differences on the first record cause program termination. Again, succeeding check failures cause error messages to be printed. |
| C. Day | A specific range of 1-28, 29, 30 or 31 is tested, depending on the month being verified. |
| D. Pilot | To date, there are 19 pilots who keep or have kept flight logs for this survey. Therefore, a range of 1-19 is tested. |
| E. Flight | Two flights in one day is not uncommon, but more than two flights are uncommon. A range of 1-3 is tested. |
| F. Block Number | No test. |

- G. CF&G Species Species code is checked against a list of
Code Number the most common species codes. If the
 species code is not found in this list,
 an error message will be printed.
- H. Day/Night This code can be a 1,2 or 3.
- I. Comments This code is tested for a range of 1-9.
- J. Sighting Type A range of 1-6 is checked.

Sighting Type = 1

- K. Number of Schools Value must be greater than 0.
- L. Total Tonnage Value must equal 0.
- M. Number of Individuals Value must equal 0.

Sighting Type = 2

- N. Number of Schools Value must equal 0.
- O. Total Tonnage Value must be greater than 0.
- P. Number of Individuals Value must equal 0.

Sighting Type = 3

- Q. Number of Schools Value must equal 0.
- R. Total Tonnage Value must equal 0.
- S. Number of Individuals Value must be greater than 0.

Sighting Type = 4 or 5

- T. Number of Schools Value must be greater than 0.
- U. Total Tonnage Value must be greater than 0.
- V. Number of Individuals Value must equal 0.

Sighting Type = 6

W. Number of Schools Value must equal 0.

X. Total Tonnage Value must equal 0.

Y. Number of Individuals Value must equal 0.

Z. Species Code Test for 1.
 Table Number

! SEDIT.COM

! ET DEFAULT UDISK6:[NM42]

UN [NM42.EDIT]SEDI

3.D

83 ! YEAR

01 ! BEGINNING MONTH

OST

XIT

PROGRAM: SEDIT.FOR

THIS PROGRAM VERIFIES THE SIGHTING DATA.

ORIGINAL PROGRAMMER: JOE CARUSO

REVISED FOR THE VAX BY SUSAN IACOMETTI IN MAY 1982.

REVISED FOR NEW DATA FORMAT BY CINDY MEYER IN OCTOBER 1982.

INTRINSIC MOD

COMMON/AA/FISHNO(81),SPECIE,MATCH

DIMENSION ERROR(14),DAYS(12),ERTEST(14),MESSAGE(14)

CHARACTER ERROR*6,MESSAGE*6,KEY*23,IKEY*23

CHARACTER*30 AFILE

INTEGER YEAR, DAY, PILOT, FLIGHT, BLOCK, SPECIE, DANITE, COMENT, STYPE,

*TOTAL, TABLE, CARD, ERTEST, DAYS, ERR, YR, FLT,

*FISHNO

INITIALIZE VARIABLES AND ARRAYS

DATA ERROR(1)/' YEAR '/,ERROR(2)/' MONTH'/,ERROR(3)/' DAY '/,
* ERROR(4)/' PILOT '/,ERROR(5)/' FLIGHT'/,ERROR(6)/' BLOCK'/,
* ERROR(7)/' SPECIE'/,ERROR(8)/' DANITE'/,ERROR(9)/' COMENT'/,
* ERROR(10)/' STYPE'/,ERROR(11)/' NOSC '/,ERROR(12)/' TOTAL '/,
* ERROR(13)/' IND '/,
* ERROR(14)/' TABLE'/

DATA DAYS/31,28,31,30,31,30,31,31,30,31,30,31/

DATA FISHNO/1,2,3,4,5,6,8,19,40,50,51,55,91,92,95,100,110,

* 130,150,156,167,176,180,184,292,300,335,349,400,402,
* 420,446,550,552,711,800,814,960,998,1111,1112,1113,
* 1114,1115,1116,1117,1118,1119,1120,1121,1122,1123,1124,
* 1125,1126,1127,1128,1129,1130,1131,1132,1133,1134,1135,
* 1136,1137,1138,1139,1141,1142,1144,2000,2001,
* 2002,2003,2004,2050,3000,3001,9001,9999/

READ DATA FILE NAME

PRINT*,'SIGHT FILE?'

READ(5,88)AFILE

88 FORMAT(A)

OPEN THE DATA FILE

OPEN(UNIT=11,TYPE='OLD',NAME=AFILE,READONLY)

INITIALIZE THE 'NO ERROR' INDICATOR

NOERR=0

CARD=0

CLEAR THE ERROR TEST AND MESSAGE ARRAYS

DO 10 I=1,14

ERTEST(I)=0

10 MESSAGE(I)=' '

N=0

READ THE TEST YEAR AND MONTH

PRINT*, 'DESIRED YEAR?'

READ(5, 1) YR

PRINT*, 'BEGINNING MONTH?'

READ(5, 1) MTH

READ THE FIRST SIGHTING DATA RECORD ON DISK

READ(11, 2) KEY, DANITE,

*COMENT, STYPE, NOSC, TOTAL, IND, TABLE

READ(KEY, 8) YEAR, MONTH, DAY, PILOT, FLIGHT, BLOCK, SPECIE

TEST THE YEAR AND MONTH

IF(YEAR.NE.YR) GO TO 99

IF(MONTH.NE.MTH) GO TO 111

WHEN A LEAP YEAR OCCURS, SET THE NUMBER OF DAYS IN FEBRUARY TO 29.

14 IF((MONTH.EQ.2).AND.(MOD(YEAR,4).EQ.0)) DAYS(2)=29
GO TO 12

READ THE NEXT SIGHTING RECORD ON DISK.

11 READ(11, 2, END=97) KEY,
*DANITE, COMENT, STYPE, NOSC, TOTAL, IND, TABLE
READ(KEY, 8) YEAR, MONTH, DAY, PILOT, FLIGHT, BLOCK, SPECIE

INCREMENT THE CARD COUNTER

12 CARD=CARD+1
IF(KEY.EQ.IKEY) GO TO 13
IF(YEAR.NE.YR) ERTEST(1)=1
IF(MONTH.NE.MTH) GO TO 98
201 IKEY=KEY

TEST THE NUMBER OF DAYS IN THE MONTH

13 IF((DAY.LT.1).OR.(DAY.GT.DAYS(MONTH))) ERTEST(3)=1

TEST THE PILOT CODE

IF((PILOT.LT.1).OR.(PILOT.GT.19)) ERTEST(4)=1
IF(PILOT.EQ.90) ERTEST(4)=0

TEST THE FLIGHT NUMBER

IF((FLIGHT.LT.1).OR.(FLIGHT.GT.3)) ERTEST(5)=1

TEST THE DAY/NIGHT CODE

IF((DANITE.LT.1).OR.(DANITE.GT.3)) ERTEST(8)=1

TEST THE COMMENT CODE

IF((COMENT.LT.1).OR.(COMENT.GT.9)) ERTEST(9)=1

TEST THE RANGE OF THE SIGHTING TYPE

IF((STYPE.LT.1).OR.(STYPE.GT.6)) ERTEST(10)=1

TEST FOR BLANK SPECIES CODE

IF(SPECIE.LT.1) ERTEST(7)=1

TEST THE SPECIES SIGHTED.

CALL SEARCH

IF(MATCH.EQ.2) ERTEST(7)=1

DEPENDING ON THE STYPE, BRANCH TO THE APPROPRIATE STATEMENT

GO TO (15,16,17,18,19,20) STYPE

TEST SIGHTING TYPE 1.

15 IF (COMENT.EQ.0) ERTEST(9)=1
IF (NOSC.EQ.0) ERTEST(11)=1
IF (TOTAL.NE.0) ERTEST(12)=1
IF (IND.NE.0) ERTEST(13)=1
GO TO 26

TEST SIGHTING TYPE 2

16 IF (TOTAL.EQ.0) ERTEST(12)=1
IF (NOSC.NE.0) ERTEST(11)=1
IF (IND.NE.0) ERTEST(13)=1
GO TO 26

TEST SIGHTING TYPE 3.

17 IF (IND.EQ.0) ERTEST(13)=1
IF (NOSC.NE.0) ERTEST(11)=1
IF (TOTAL.NE.0) ERTEST(12)=1
GO TO 26

TEST SIGHTING TYPE 4.

18 IF (NOSC.EQ.0) ERTEST(11)=1
IF (TOTAL.EQ.0) ERTEST(12)=1
IF (IND.NE.0) ERTEST(13)=1
GO TO 26

TEST SIGHTING TYPE 5.

19 IF (NOSC.EQ.0) ERTEST(11)=1
IF (TOTAL.EQ.0) ERTEST(12)=1


```
IF (IND.NE.0) ERTEST(13)=1
GO TO 26
```

TEST SIGHTING TYPE 6.

```
20 IF (NOSC.NE.0) ERTEST(11)=1
IF (TOTAL.NE.0) ERTEST(12)=1
IF (IND.NE.0) ERTEST(13)=1
GO TO 26
```

THE SPECIES TABLE NUMBER AT THIS TIME SHOULD ALWAYS BE 1.

```
26 IF (TABLE.NE.1) ERTEST(21)=1
```

FILL THE ERROR MESSAGE ARRAY

```
28 DO 29 I=1,14
IF (ERTEST(I).EQ.0) GO TO 29
NOERR=1
ERR=1
MESSAGE(I)=ERROR(I)
29 CONTINUE
```

TEST ERROR CODE. IF NO ERRORS, BRANCH TO READ ANOTHER CARD.

```
IF (ERR.EQ.1) GO TO 30
GO TO 11
```

IF ERRORS WERE FOUND, PRINT RECORD WITH PROPER ERROR MESSAGES.

```
30 WRITE(6,5) YEAR,MONTH,DAY,PILOT,FLIGHT,BLOCK,SPECIE,DANITE,
*COMENT,STYPE,NOSC,TOTAL,IND,TABLE,CARD,
*(MESSAGE(J),J=1,14)
```

CLEAR THE ERTEST AND MESSAGE ARRAYS.

```
DO 31 I=1,14
ERTEST(I)=0
31 MESSAGE(I)=' '
ERR=0
GO TO 11
```

*****FORMAT STATEMENTS *****

```
1 FORMAT(7X,I2)
2 FORMAT(A23,2I1,I2,I4,I9,I4,I1)
8 FORMAT(3I2,I4,I1,2X,I6,I4)
5 FORMAT('0',3I3,I5,I2,I7,I5,2I2,I3,I5,I10,I5,I2, /
*' THIS RECORD HAS A POSSIBLE ERROR ON IT. THE RECORD NO. = ',I5,
*/22A6)
6 FORMAT('0','THE YEAR READ = ',I2,' AND IS NOT EQUAL TO THE YEAR
*ON THE RECORD WHICH = ',I2)
7 FORMAT('0','THE MONTH READ = ',I2,' AND IS NOT EQUAL TO THE MONTH
* ON THE RECORD WHICH = ',I2)
9 FORMAT('1','***** THE SIGHTING VERIFICATION PROGRAM, SEDIT, HAS E
```


*EXECUTED SUCCESSFULLY. NO ERRORS WERE DETECTED. *****')

50 FORMAT('1','***** THE SIGHTING VERIFICATION PROGRAM, SEDIT, HAS E
*EXECUTED. SOME POSSIBLE ERRORS WERE DETECTED. *****')

PRINT MESSAGE AT END OF PROGRAM

97 IF(NOERR.GT.0) WRITE(6,50)
IF(NOERR.EQ.0) GO TO 100
GO TO 101

INCREMENT MONTH COUNTER

98 IF(MONTH.NE.(MTH+1))GO TO 211
MTH=MTH+1
CARD=CARD-1
GO TO 14

SET ERROR CODE IF ERROR IN MONTH

211 ERTEST(2)=1
GO TO 201

WRITE ERROR MESSAGE FOR ERROR IN MONTH

111 WRITE(6,7)MTH,MONTH
GO TO 101

WRITE ERROR MESSAGE FOR ERROR IN YEAR

99 WRITE(6,6) YR, YEAR
GO TO 101
100 WRITE(6,9)
101 CONTINUE
STOP
END

SEARCH: A SUBROUTINE

SUBROUTINE SEARCH

THE ARRAY FISHNO IS SEARCHED TO SEE IF SPECIES CODE IS VALID.
ELEMENTS IN ARRAY 'FISHNO' ARE ORDERED IN AN ASCENDING MANNER

COMMON/AA/FISHNO(81),SPECIE,MATCH
INTEGER FISHNO,SPECIE,X,Y,HIGH,LOW

THE LOWEST NUMBER IS ALWAYS SET TO ZERO

LOW=0

THE HIGHEST NUMBER IS ALWAYS SET TO ONE MORE THAN THE NUMBER OF
ELEMENTS IN THE ARRAY

HIGH=82

MATCH=1

SEARCH THE ARRAY TO SEE IF SPECIES NUMBER IS VALID

```
2  IF((HIGH-LOW).LT.2) GO TO 3
   X=(HIGH-LOW)/2
   Y=LOW+X
   IF(SPECIE.EQ.FISHNO(Y)) GO TO 4
   IF(SPECIE.GT.FISHNO(Y)) LOW=Y
   IF(SPECIE.LT.FISHNO(Y)) HIGH=Y
   GO TO 2
3  MATCH=2
4  RETURN
   END
```


EDISK

Effort Data Program

Purpose

The effort program, EDISK, checks the block numbers recorded and transforms old 3-digit block numbers to a 6-digit block number. Records with block number errors are flagged for correction.

Input

EDISK uses four sets of input. The first set is the raw data. The raw data is in fixed-length records created by a keypuncher, with several records per flight. Set two is a file of 3-digit block numbers invented by the California Department of Fish and Game. The next input set is the revised 6-digit block numbers corresponding to the Fish and Game numbers. The last set contains all possible block numbers through which pilots might fly. Sets 2-4 are sorted in ascending order. Since sets 2-4 must always be the same, the program automatically opens their files and reads them in.

The end result of the program is a disk file of effort data that is acceptable to the effort verification program. Records will be variable-length, with one per flight.


```
$ !  
$ ! EDISK.COM  
$ !  
$SET DEFAULT UDISK6:[NM42]  
$RUN [NM42.EDIT]EDISK  
EFFORT83.DAT  
E83.D  
$COST  
$EXIT
```


PROGRAM: EDISK.FOR

THIS PROGRAM CREATES A DISK FILE OF THE AERIAL MARINE RESOURCES
MONITORING SYSTEM EFFORT CARD IMAGES. IT WILL CHANGE ANY OLD THREE
DIGIT BLOCK NUMBER TO THE NEW SIX-DIGIT NUMBER AND THEN WRITE THE
CARD IMAGE ON DISK.

CHANGES THE FILE TO VARIABLE LENGTH RECORDS, ONE RECORD PER FLIGHT.

ORIGINAL PROGRAMMER: JOE CARUSO

REVISED FOR THE VAX SYSTEM BY SUSAN IACOMETTI IN APRIL 1982.

REVISED FOR NEW DATA FORMAT BY CINDY MEYER IN OCTOBER 1982.

COMMON/AA/ ONEWBK(778),BK(8),BLK(160)

COMMON/BLOCK/ CFGBK(345),REVBK(345),BLOCK, YEAR, MONTH, DAY,

*PILOT, NO

CHARACTER*30 INFILE,OUTFILE

INTEGER BLOCK,CFGBK,REVBK,BLK, YEAR, MONTH, DAY, PILOT,

*BK, SEQ, ONEWBK, MAX

CHARACTER KEY*11, DATA*19, OKEY*11, ODATA*19

READ IN DATA FILE NAMES

200 FORMAT(A)

PRINT*, 'EFFORT FILE?'

READ(5,200)INFILE

PRINT*, 'OUTPUT EFFORT FILE?'

READ(5,200)OUTFILE

OPEN ALL FILES USED BY THE PROGRAM

OPEN(UNIT=25, TYPE='OLD', NAME=INFILE, READONLY)

OPEN(UNIT=12, TYPE='NEW', NAME=OUTFILE, CARRIAGECONTROL='LIST',
*FORM='FORMATTED', RECORDTYPE='VARIABLE', RECL=993)

OPEN(UNIT=9, TYPE='OLD', NAME='[NM42.EDIT]BLOCK.DAT', READONLY)

OPEN(UNIT=15, TYPE='OLD', NAME='[NM42.EDIT]ORB.DAT', READONLY)

READ THE OLD BLOCK NUMBERS INTO THE ARRAY "CFGBK"

READ THE NEW BLOCK NUMBERS INTO THE ARRAY "REVBK"

READ(9,1) (CFGBK(I),REVBK(I),I=1,345)

READ ALL POSSIBLE SIX-DIGIT BLOCK NUMBERS INTO ARRAY "ONEWBK"

READ(15,2) (ONEWBK(I),I=1,778)

INITIALIZE CARD COUNTER

NO=0

READ AN EFFORT CARD

READ(25,3,END=99,ERR=98)KEY, DATA, BK, SEQ


```
C INITIALIZE BLOCK COUNTER
C
20  MAX=0
C INCREMENT CARD COUNTER
C
  NO=NO+1
C SET OKEY AND ODATA VALUES
C
  OKEY=KEY
  ODATA=DATA
  READ (KEY, 7) YEAR, MONTH, DAY, PILOT
  DO 30 L=1, 11
30  IF (OKEY(L:L).EQ.' ') OKEY(L:L)='0'
  DO 40 L=1, 19
40  IF (ODATA(L:L).EQ.' ') ODATA(L:L)='0'
C READ BLOCKS FROM FIRST CARD INTO BLOCK ARRAY AND
C CHANGE THEM TO THE 6-DIGIT BLOCK NUMBERS
C
  DO 50 I=1, 8
  IF (BK(I).EQ.0) GO TO 55
  MAX=MAX+1
  BLOCK=BK(I)
  IF ((BLOCK.GT.990).AND.(BLOCK.LT.1000)) GO TO 50
  CALL SEARCH(I)
50  BLK(I)=BK(I)
C READ NEXT EFFORT CARD
C
55  READ(25, 3, END=99, ERR=98) KEY, DATA, BK, SEQ
C IF IT IS A NEW FLIGHT, WRITE THE OLD FLIGHT AND RETURN TO
C INITIALIZE THE COUNTERS AND SET OKEY AND ODATA.
C
  IF (SEQ.EQ.1) THEN
    M=MAX
    WRITE(12, 4) OKEY, ODATA, M, (BLK(I), I=1, M)
    GO TO 20
  END IF
C IF IT IS THE SAME FLIGHT, CONTINUE PUTTING BLOCKS INTO THE
C BLOCK ARRAY AND TRANSLATING THEM TO 6-DIGIT BLOCK VALUES.
C
  DO 300 I=1, 8
  IF (BK(I).EQ.0) GO TO 301
  MAX=MAX+1
  BLOCK=BK(I)
  IF ((BLOCK.GT.990).AND.(BLOCK.LT.1000)) GO TO 300
  CALL SEARCH(I)
300  BLK(MAX)=BK(I)
C RETURN TO READ ANOTHER CARD
C
```



```
301 GO TO 55
C
C***** FORMAT STATEMENTS *****
C
1 FORMAT(2X,I3,2X,I6)
2 FORMAT(2X,I6)
3 FORMAT(A11,A19,8I6,I2)
4 FORMAT(A11,A19,I3.3,(M)I6.6)
5 FORMAT(1H1,'***** EDISK HAS ENCOUNTERED ERRORS DURING EXECUTION.
* THE PROGRAM HAS EXITED ABNORMALLY *****')
6 FORMAT(1H1,'***** EDISK HAS EXECUTED SUCCESSFULLY *****')
7 FORMAT(3I2,I4)
C
C PRINT THE FINAL MESSAGE OF THE PROGRAM
C
98 WRITE(6,5)
GO TO 100
99 M=MAX
WRITE(12,4) OKEY, ODATA, M, (BLK(I), I=1, M)
WRITE(6,6)
100 CLOSE(UNIT=12)
STOP
END
C
C SEARCH: A SUBROUTINE
C
SUBROUTINE SEARCH(I)
C
C PERFORMS A BINARY SEARCH OF THE BLOCK ARRAYS AND ASSIGNS A
C 6-DIGIT BLOCK VALUE FOR A 3-DIGIT BLOCK VALUE
C
COMMON/AA/ ONEWBK(778), BK(8), BLK(160)
COMMON/BLOCK/ CFGBK(345), REVBK(345), BLOCK, YEAR, MONTH, DAY
*, PILOT, NO
INTEGER BLOCK, CFGBK, REVBK, YEAR, DAY, PILOT, X, Y, HIGH, ONEWBK
*, BK
C
C SET THE BOTTOM AND TOP OF THE SEARCH RANGE
C
LOW=0
HIGH=346
IF(BLOCK.GT.989) GO TO 6
C
C SEARCH THE ARRAY OF 3-DIGIT BLOCK NUMBERS TO SEE IF THE BLOCK IS THERE
C
2 IF((HIGH-LOW).LT.2) GO TO 3
X=(HIGH-LOW)/2
Y=LOW+X
IF(BLOCK.EQ.CFGBK(Y)) GO TO 4
IF(BLOCK.GT.CFGBK(Y)) LOW=Y
IF(BLOCK.LT.CFGBK(Y)) HIGH=Y
GO TO 2
6 HIGH=779
C
```



```
C SEARCH THE COMPLETE ARRAY OF 6-DIGIT BLOCK NUMBERS FOR THE
C DESIRED BLOCK NUMBER
C
5 IF((HIGH-LOW).LT.2)GO TO 3
  X=(HIGH-LOW)/2
  Y=LOW+X
  IF(BLOCK.EQ.ONEWBK(Y))GO TO 7
  IF(BLOCK.GT.ONEWBK(Y))LOW=Y
  IF(BLOCK.LT.ONEWBK(Y))HIGH=Y
  GO TO 5
3 WRITE(6,1)NO,YEAR,MONTH,DAY,PILOT,BLOCK
1 FORMAT(1H ,'CARD NO. = ',I5,' WITH A "KEY" EQUAL TO ',3I2,' AND
* PILOT CODE = ',I4,' HAS A BLOCK NO. = ',I6,' WHICH WAS NOT FOU
*ND IN THE MASTER LIST.')
```

RETURN

```
4 BK(I)=REVBK(Y)
7 RETURN
END
```


FMSEDISK

Effort Data Program

Purpose

The effort program, FMSEDISK, checks the block numbers recorded and transforms old 3-digit block numbers to 6-digit block numbers. Records with block number errors are flagged for correction.

Input

FMSEDISK uses four sets of input. The first set is the raw data. The raw data is in variable length records created by FMS, with one record per flight. Set two is a disk file with a list of block numbers originating from the California Department of Fish and Game. Next is a file containing revised Fish and Game block numbers. The last set is a file of all possible block numbers through which the pilots might fly. Sets 2-4 are sorted in ascending order. Sets 2-4 must always remain the same, so the program automatically opens the files and reads the block numbers in.

The end result of the program is a disk file of effort data that is acceptable to the effort verification program. Records are variable length, with one record per flight.


```
$ !  
$ ! FMSEDISK.COM  
$ !  
$SET DEFAULT UDISK6:[NM42]  
$RUN [NM42.EDIT]FMSEDISK  
EFFORT83.DAT  
E83.D  
$COST  
$EXIT
```



```
C
C PROGRAM:  FMSEDISK.FOR
C
C THIS PROGRAM CREATES A DISK FILE OF THE AERIAL MARINE RESOURCES
C MONITORING SYSTEM EFFORT RECORD IMAGES.  IT WILL CHANGE ANY OLD THREE
C DIGIT BLOCK NUMBER TO THE NEW SIX-DIGIT NUMBER AND THEN WRITE THE
C CARD IMAGE ON DISK.
C CHANGES THE FILE TO VARIABLE LENGTH RECORDS, ONE RECORD PER FLIGHT.
C
C ORIGINAL PROGRAMMER: JOE CARUSO
C REVISED FOR THE VAX SYSTEM BY SUSAN IACOMETTI IN APRIL 1982.
C REVISED FOR NEW DATA FORMAT BY CINDY MEYER IN OCTOBER 1982.
C
C FOR USE WITH DATA ENTERED WITH FORMS MANAGEMENT.
C
C COMMON/AA/ ONEWBK(778),BK(160),BLK(160)
C COMMON/BLOCK/ CFGBK(345),REVBK(345),BLOCK, YEAR, MONTH, DAY,
C *PILOT, NO
C CHARACTER*30 INFILE,OUTFILE
C INTEGER BLOCK,CFGBK,REVBK,BLK, YEAR, MONTH, DAY, PILOT,
C *BK, SEQ, ONEWBK, MAX
C CHARACTER KEY*11, DATA*19
C
C READ IN DATA FILE NAMES
C
C 200 FORMAT(A)
C PRINT*, 'EFFORT FILE?'
C READ(5,200)INFILE
C PRINT*, 'OUTPUT EFFORT FILE?'
C READ(5,200)OUTFILE
C
C OPEN ALL FILES USED BY THE PROGRAM
C
C OPEN(UNIT=25, TYPE='OLD', NAME=INFILE, READONLY,
C *FORM='FORMATTED', RECORDTYPE='VARIABLE', RECL=993)
C OPEN(UNIT=12, TYPE='NEW', NAME=OUTFILE, CARRIAGECONTROL='LIST',
C *FORM='FORMATTED', RECORDTYPE='VARIABLE', RECL=993)
C OPEN(UNIT=9, TYPE='OLD', NAME=' [NM42.EDIT]BLOCK.DAT', READONLY)
C OPEN(UNIT=15, TYPE='OLD', NAME=' [NM42.EDIT]ORB.DAT', READONLY)
C
C READ THE OLD BLOCK NUMBERS INTO THE ARRAY "CFGBK"
C READ THE NEW BLOCK NUMBERS INTO THE ARRAY "REVBK"
C
C READ(9,1) (CFGBK(I),REVBK(I),I=1,345)
C
C READ ALL POSSIBLE SIX-DIGIT BLOCK NUMBERS INTO ARRAY "ONEWBK"
C
C READ(15,2) (ONEWBK(I),I=1,778)
C
C INITIALIZE CARD COUNTER
C
C NO=0
C
C READ AN EFFORT CARD
C
```



```
20 READ(25,3,END=99,ERR=98)KEY,DATA,M,(BK(I),I=1,M)
   READ(KEY,7)YEAR,MONTH,DAY,PILOT
C
C INCREMENT CARD COUNTER
C
   NO=NO+1
C
C CHANGE BLOCKS FROM RECORD
C INTO THE 6-DIGIT BLOCK NUMBERS
C
   DO 50 I=1,M
   BLOCK=BK(I)
   IF ((BLOCK.GT.990).AND.(BLOCK.LT.1000)) GO TO 50
   CALL SEARCH(I)
50  BLK(I)=BK(I)
   WRITE(12,4)KEY,DATA,M,(BLK(I),I=1,M)
   GO TO 20
C
C***** FORMAT STATEMENTS *****
C
1  FORMAT(2X,I3,2X,I6)
2  FORMAT(2X,I6)
3  FORMAT(A11,A19,I3,(M)I6)
4  FORMAT(A11,A19,I3.3,(M)I6.6)
5  FORMAT(1H1,'***** EDISK HAS ENCOUNTERED ERRORS DURING EXECUTION.
   * THE PROGRAM HAS EXITED ABNORMALLY *****')
6  FORMAT(1H1,'***** EDISK HAS EXECUTED SUCCESSFULLY *****')
7  FORMAT(3I2,I4)
C
C PRINT THE FINAL MESSAGE OF THE PROGRAM
C
98  WRITE(6,5)
   GO TO 100
99  WRITE(6,6)
100 CLOSE(UNIT=12)
   STOP
   END
C
C SEARCH: A SUBROUTINE
C
   SUBROUTINE SEARCH(I)
C
C PERFORMS A BINARY SEARCH OF THE BLOCK ARRAYS AND ASSIGNS A
C 6-DIGIT BLOCK VALUE FOR A 3-DIGIT BLOCK VALUE
C
   COMMON/AA/ ONEWBK(778),BK(160),BLK(160)
   COMMON/BLOCK/ CFGBK(345),REVBK(345),BLOCK,YEAR,MONTH,DAY
   *,PILOT,NO
   INTEGER BLOCK,CFGBK,REVBK,YEAR,DAY,PILOT,X,Y,HIGH,ONEWBK
   *,BK
C
C SET THE BOTTOM AND TOP OF THE SEARCH RANGE
C
   LOW=0
```



```
HIGH=346  
IF(BLOCK.GT.989)GO TO 6
```

SEARCH THE ARRAY OF 3-DIGIT BLOCK NUMBERS TO SEE IF THE BLOCK IS THERE

```
2 IF((HIGH-LOW).LT.2) GO TO 3  
X=(HIGH-LOW)/2  
Y=LOW+X  
IF(BLOCK.EQ.CFGBK(Y))GO TO 4  
IF(BLOCK.GT.CFGBK(Y))LOW=Y  
IF(BLOCK.LT.CFGBK(Y))HIGH=Y  
GO TO 2  
6 HIGH=779
```

SEARCH THE COMPLETE ARRAY OF 6-DIGIT BLOCK NUMBERS FOR THE
DESIRED BLOCK NUMBER

```
5 IF((HIGH-LOW).LT.2)GO TO 3  
X=(HIGH-LOW)/2  
Y=LOW+X  
IF(BLOCK.EQ.ONEWBK(Y))GO TO 7  
IF(BLOCK.GT.ONEWBK(Y))LOW=Y  
IF(BLOCK.LT.ONEWBK(Y))HIGH=Y  
GO TO 5
```

```
3 WRITE(6,1)NO,YEAR,MONTH,DAY,PILOT,BLOCK
```

```
1 FORMAT(1H,'CARD NO. = ',I5,' WITH A "KEY" EQUAL TO ',3I2,' AND  
* PILOT CODE = ',I4,' HAS A BLOCK NO. = ',I6,' WHICH WAS NOT FOU  
*ND IN THE MASTER LIST.')
```

```
RETURN
```

```
4 BK(I)=REVBK(Y)
```

```
7 RETURN
```

```
END
```


EEDIT

Effort Data Verification Program

Purpose

The effort verification program, EEDIT, checks the effort data records and flags, for correction, any records which may have errors on them.

Input

The input needed for EEDIT consists of the effort data, a year and a month. The year and month are the year and month in which the data begins.

Verification Procedures

The verification tests used in EEDIT will now be discussed.

- | | |
|-----------|--|
| A. Year | <u>Input Data</u> - 2 digits, punched in columns 8 and 9.
As the year is read from the effort data file, the first record is checked to be sure the proper file has been accessed. Any deviations cause program termination. Each succeeding record check failure causes the record to be printed. |
| B. Month | <u>Input Data</u> - 2 digits punched in columns 8 and 9 of the next line of the command file.
The first time the month is read it is tested to be sure the proper file has been opened. Any deviations cause program termination. Each succeeding record test that fails causes the record to be printed. |
| C. Day | Depending on the month involved, a specific range is tested (i.e. 1-28, 29, 30 or 31). Provisions have been made for a leap year. |
| D. Pilot | The number of pilots is a known quantity. A range of 1-19 is tested. |
| E. Flight | The average number of flights made by one pilot in one day is 1.5. Two flights are not uncommon, but three or more flights |

are uncommon.

A range of 1-3 is tested.

F. Day/Night

The day or night variable has only 3 possibilities, so a range of 1-3 is tested.

G. Time Out
Time In

Three tests are associated with these two variables. First, each is checked for blanks or zeros; if either is a blank, it is flagged and the next two tests are skipped. Next, Time Out is tested to make sure it is smaller than Time In. Third, the length of the flight is found and tested. Most flights are 3-5 hours in length, but some will be greater. Any flight length greater than 15 hours is flagged.

H. Plane

The number of planes the pilots use are relatively constant. A range of 1-48 is tested.

I. Plane/Pilot

Some planes are flown only by one pilot. Since this information is known, a series of plane-pilot checks are initiated.

J. Comments

There are two different codes which can be placed in this field, therefore a range of 1-2 is used.

K. Block Numbers

Each block number is tested against the next block number. The absolute difference, except for special cases, must be 1001, 1000, 999 or 1. (Refer to "Block Area Numbering Method".) The special cases are the unknown effort numbers 991, 992, 993, 994 and 995. Another special case is the overland flight block number 999. 990 is also classed as unknown effort. For further information, refer to Effort Usage: Unknown Effort.


```
## !  
## ! EEDIT.COM  
## !  
##SET DEFAULT UDISK6:[NM42]  
##RUN [NM42.EDIT]EEDIT  
E83.D  
      83      ! YEAR  
      01      ! BEGINNING MONTH  
##COST  
##EXIT
```



```
C
C PROGRAM:  EEDIT.FOR
C
C      INTRINSIC MOD
C
C      THIS PROGRAM VERIFIES THE AERIAL MARINE RESOURCES MONITORING
C      SYSTEM EFFORT RECORDS.
C
C ORIGINAL PROGRAMMER: JOE CARUSO
C REVISED FOR THE VAX BY SUSAN IACOMETTI IN MAY 1982.
C REVISED FOR NEW DATA FORMAT BY CINDY MEYER IN OCTOBER 1982.
C
C      DIMENSION ERROR(10), DAYS(12), ERTEST(13), BK(160), SHEET(7),
C      *MESSAGE(10)
C      CHARACTER ERROR*6, MESSAGE*6
C      CHARACTER*30 AFILE
C      REAL INTIME, OUTIME
C      INTEGER YEAR, DAY, PILOT, FLIGHT, DANITE, PLANE, COMENT, BK, SEQ, NUMBLK,
C      *DAYS, ERTEST, DYEAR, DMONTH, ERR, CARD, YR, BLOCK, BLK, SHEET, FLT
C
C INITIALIZE VARIABLES
C
C      DATA ERROR(1)/' YEAR '/, ERROR(2)/' MONTH'/, ERROR(3)/' DAY  '/,
C      *ERROR(4)/' PILOT '/, ERROR(5)/' FLIGHT'/, ERROR(6)/' DANITE'/,
C      *ERROR(7)/' OUTIME'/, ERROR(8)/' INTIME'/, ERROR(9)/' PLANE'/,
C      *ERROR(10)/' COMENT'/
C      DATA DAYS/31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31/
C      DATA SHEET/990, 991, 992, 993, 994, 995, 999/
C
C READ EFFORT FILE NAME
C
C      88 FORMAT(A)
C      PRINT*, 'EFFORT FILE?'
C      READ(5, 88) AFILE
C
C OPEN DATA FILE
C
C      OPEN(UNIT=12, TYPE=' OLD', NAME=AFILE, READONLY)
C
C      INITIALIZE THE 'NO ERROR' VARIABLE.
C
C      NOERR=0
C
C      INITIALIZE THE CARD COUNTER AND ARRAYS MESSAGE AND ERTEST.
C
C      BLOCK=0
C      CARD=0
C      DO 10 I=1, 13
C      ERTEST(I)=0
C 10  MESSAGE(I)='
C
C READ YEAR AND MONTH
C
C      PRINT*, 'WHAT YEAR?'
C      READ(5, 4) YR
```



```
PRINT*, 'BEGINNING MONTH?'  
READ(5, 4) MTH
```

```
C  
C READ AN EFFORT RECORD  
C
```

```
READ(12, 1) YEAR, MONTH, DAY, PILOT, FLIGHT, DANITE, OUTIME, INTIME, PLANE,  
*COMENT, NUMBLK, (BK(I), I=1, NUMBLK)
```

```
C  
C CHECK THE YEAR AND MONTH FOR AGREEMENT WITH THE DATA  
C
```

```
IF(YEAR.NE.YR) GO TO 99  
IF(MONTH.NE.MTH) GO TO 111
```

```
C  
C TEST FOR LEAP YEAR  
C
```

```
110 IF((MONTH.EQ.2).AND.(MOD(YEAR, 4).EQ.0)) DAYS(2)=29  
GO TO 21
```

```
C  
C READ THE NEXT RECORD ON DISK  
C
```

```
11 READ(12, 1, END=100) YEAR, MONTH, DAY, PILOT, FLIGHT, DANITE, OUTIME,  
*INTIME, PLANE, COMENT, NUMBLK, (BK(I), I=1, NUMBLK)  
21 CARD=CARD+1
```

```
C  
C FILL THE DUMMY VARIABLES  
C
```

```
DYEAR=YEAR  
DMONTH=MONTH  
BLOCK=0
```

```
C  
C TEST THE YEAR AND MONTH FOR BLANKS  
C
```

```
IF(YEAR.NE.YR) ERTEST(1)=1  
IF(MONTH.NE.MTH) GO TO 98
```

```
C  
C TEST THE NUMBER OF DAYS RECORDED  
C
```

```
201 IF((DAY.LT.1).OR.(DAY.GT.DAYS(MONTH))) ERTEST(3)=1
```

```
C  
C TEST THE PILOT CODE RECORDED  
C
```

```
IF((PILOT.LT.1).OR.(PILOT.GT.19)) ERTEST(4)=1  
IF(PILOT.EQ.90) ERTEST(4)=0
```

```
C  
C TEST THE FLIGHT NUMBER RECORDED  
C
```

```
IF((FLIGHT.LT.1).OR.(FLIGHT.GT.3)) ERTEST(5)=1
```

```
C  
C TEST THE DAY OR NIGHT CODE RECORDED  
C
```

```
IF((DANITE.LT.1).OR.(DANITE.GT.3)) ERTEST(6)=1
```

```
C  
C TEST THE RECORDED OUTIME AND INTIME  
C
```

```
IF ((INTIME.EQ.0).AND.(OUTIME.EQ.0)) GO TO 22
```



```
IF (INTIME.EQ.0) ERTEST(8)=1
IF (OUTIME.EQ.0) ERTEST(7)=1
IF (INTIME.LE.OUTIME) ERTEST(8)=1
```

```
TEST THE LENGTH OF THE FLIGHT
```

```
IF((INTIME-OUTIME).GT.15)ERTEST(7)=1
```

```
TEST THE PLANE CODE RECORDED
```

```
22 IF((PLANE.LT.1).OR.(PLANE.GT.48)) ERTEST(9)=1
IF(PLANE.EQ.99) ERTEST(9)=0
```

```
TEST THE PLANE AND THE PILOT FLYING THE PLANE
```

```
IF((PLANE.EQ.37).AND.(PILOT.NE.15)) ERTEST(4)=1
IF((PLANE.EQ.38).AND.(PILOT.NE.16)) ERTEST(4)=1
IF((PLANE.EQ.39).AND.(PILOT.NE.17)) ERTEST(4)=1
IF((PLANE.EQ.40).AND.(PILOT.NE.17)) ERTEST(4)=1
IF((PLANE.EQ.41).AND.(PILOT.NE.18)) ERTEST(4)=1
IF((PLANE.EQ.42).AND.(PILOT.NE.11)) ERTEST(4)=1
IF((PLANE.EQ.43).AND.((PILOT.NE.16).AND.(PILOT.NE.15)))
* ERTEST(4)=1
```

```
IF((PLANE.EQ.44).AND.(PILOT.NE.11))ERTEST(4)=1
IF((PLANE.EQ.45).AND.(PILOT.NE.11))ERTEST(4)=1
IF((PLANE.EQ.46).AND.(PILOT.NE.15)) ERTEST(4)=1
IF((PLANE.EQ.47).AND.(PILOT.NE.16)) ERTEST(4)=1
IF((PLANE.EQ.48).AND.(PILOT.NE.19)) ERTEST(4)=1
```

```
TEST THE COMMENT CODE
```

```
IF((COMENT.LT.1).OR.(COMENT.GT.2)) ERTEST(10)=1
```

```
TEST THE FLIGHT PATH
```

```
DO 16 I=1,NUMBLK-1
BLK=ABS(BK(I)-BK(I+1))
IF(BLK.EQ.1001) GO TO 16
IF(BLK.EQ.1000) GO TO 16
IF(BLK.EQ. 999) GO TO 16
IF(BLK.EQ. 1) GO TO 16
```

```
TEST FOR UNKNOWN FLIGHT PATH
```

```
DO 24 K=I,I+1
DO 24 M=1,7
IF(BK(K).EQ.SHEET(M)) GO TO 16
24 CONTINUE
```

```
IF AN ERROR IN THE FLIGHT PATH IS FOUND, WRITE OUT WHICH CARD
HAS THE ERROR ALONG WITH THE YEAR, MONTH AND DAY.
```

```
WRITE(6,2) CARD, YEAR, MONTH, DAY, PILOT, FLIGHT, SEQ
NOERR=1
GO TO 17
```


16 CONTINUE

FILL THE ERROR MESSAGE ARRAY

17 DO 18 I=1,10
IF(ERTEST(I).EQ.0) GO TO 18
MESSAGE(I)=ERROR(I)
ERR=1
NOERR=1
18 CONTINUE
IF(ERR.EQ.1) GO TO 19
GO TO 11

THERE ARE ERRORS ON THIS RECORD. WRITE OUT THE RECORD IMAGE AND
THE LOCATION OF THE ERROR.

19 WRITE(6,3) YEAR,MONTH,DAY,PILOT,FLIGHT,DANITE,OUTIME,INTIME,PLANE,
*COMENT,SEQ,CARD,(MESSAGE(J),J=1,12)

CLEAR THE MESSAGE AND TEST ARRAYS

DO 20 I=1,10
ERTEST(I)=0
20 MESSAGE(I)=' '
ERR=0

BRANCH BACK AND READ ANOTHER CARD

GO TO 11

***** FORMAT STATEMENTS *****

1 FORMAT(3I2,I4,2I1,2F7.1,I3,I1,I3,(NUMBLK)I6)
2 FORMAT('0','THE FLIGHT PATH IS IN ERROR ON CARD NO. ',I5,
*' WITH KEY = ',3I2,I4,I1,' SEQ = ',I1)
3 FORMAT(' ',3I2,I4,2I1,2F7.1,I3,I1,I1/,' ','*** THE ERRORS ON
*RECORD NUMBER ',I5,' WERE IN: *****'/,' ',12A6)
4 FORMAT(7X,I2)
5 FORMAT('0','THE MONTH READ ON THE DATA CARD = ',I2,
*' IS INCORRECT. IT SHOULD BE = ',I2/
*1H , 'YEAR = ',I2,2X,'MONTH = ',I2,2X,'DAY = ',I2,2X,'PILOT = ',
*I4,2X,'FLIGHT = ',I1)
6 FORMAT('0','THE YEAR READ ON THE DATA CARD = ',I2,
*' IS INCORRECT. IT SHOULD BE = ',I2/
*1H , 'YEAR = ',I2,2X,'MONTH = ',I2,2X,'DAY = ',I2,2X,'PILOT = ',I4,
*2X,'FLIGHT = ',I1)
7 FORMAT('0','***** EXECUTION OF THE EFFORT VERIFICATION PROGRAM, EE
*DIT HAS ENDED. NO ERRORS WERE DETECTED. *****')
8 FORMAT('1','***** THE EFFORT VERIFICATION PROGRAM, EEDIT HAS EXEC
*UTED *****')

INCREMENT MONTH COUNTER

98 IF(MONTH.NE.(MTH+1))GO TO 211
MTH=MTH+1


```
CARD=CARD-1
GO TO 110
211 ERTEST(2)=1
GO TO 201
111 WRITE(6,5) MONTH,MTH, YEAR, MONTH, DAY, PILOT, FLIGHT
GO TO 101
99 WRITE(6,6) YEAR,YR, YEAR, MONTH, DAY, PILOT, FLIGHT
GO TO 101
100 IF(NOERR.GT.0) WRITE(6,8)
IF(NOERR.EQ.0) WRITE(6,7)
101 CONTINUE
STOP
END
```


MERGE

Effort and Sighting Data Verification Program

Purpose

The program which checks elements on the effort and sighting records is called MERGE. Internally, MERGE collates the effort and sighting records and cross-checks the matching flight elements. This last verification procedure provides an error scan not possible in the individual verification programs.

Input

The input files necessary for program MERGE are the effort and sighting data disk files.

Verification Procedures

The verification checks done on particular elements are:

- A. Key Tests the unique "key" for one effort record against the "key" of a sighting record.
 Implications: Every sighting record must have an associated effort record. However, every effort record need not have an associated sighting record.
- B. Day/Night Checks the day/night codes on the effort and sighting records for a match.
- C. Block Tests the block number on the sighting record to make sure it is one of the block numbers recorded on the effort record.

Included in the program is a summation of all records, effort and sighting. This information is then printed out as a check, and for future reference.


```
$ !  
$ ! MERGE.COM  
$ !  
$SET DEFAULT UDISK6:[NM42]  
$RUN [NM42.EDIT]MERGE  
E83.D  
S83.D  
$COST  
$EXIT
```



```
C
C PROGRAM:  MERGE.FOR
C
C      THIS PROGRAM COLLATES THE EFFORT AND SIGHTING RECORDS FOR ONE
C      MONTH AND CHECKS EACH FLIGHT MATING FOR ERRORS.
C
C ORIGINAL PROGRAMMER:  JOE CARUSO
C REVISED BY CINDY MEYER IN OCTOBER, 1982
C
C      CHARACTER KEY*11, LOCK*11, EKEY*11, SKEY*11
C      CHARACTER*30 EFILE, SFILE
C      COMMON/ONE/ TEST(160), N, BLK, MATCH
C      INTEGER BLK, COMENT, CMNT, DN, COUNT, STORE, ESEQ, EDN,
C      *      DANITE, SEQ, STYPE, DNITE, TOTAL,
C      *      TABLE, OUTIME, PLANE, SDN, SPECIE, TEST, FLTSUM, RECORD, SUM,
C      *      ECMT, SCMT, SREC
C
C      INITIALIZE THE VARIABLES
C
C      FLTSUM=0
C      RECORD=0
C      COUNT=0
C      SREC=0
C      SUM=0
C      NOEFER=0
C      NOSTER=0
C      N=0
C
C READ DATA FILE NAMES
C
C      PRINT*, 'EFFORT FILE?'
C      READ(5, 88) EFILE
C      88 FORMAT(A)
C      PRINT*, 'SIGHT FILE?'
C      READ(5, 88) SFILE
C
C OPEN THE DATA FILES
C
C      OPEN(UNIT=12, TYPE='OLD', NAME=EFILE, READONLY)
C      OPEN(UNIT=11, TYPE='OLD', NAME=SFILE, READONLY)
C
C      READ AN EFFORT RECORD
C
C      1  READ(12, 20, END=8) KEY, DANITE, OUTIME, INTIME, PLANE, COMENT,
C      *NUMBLK, (TEST(I), I=1, NUMBLK)
C      N=NUMBLK
C      RECORD=RECORD+1
C
C      STORE THE TEST VARIABLES
C
C      EKEY=KEY
C      EDN=DANITE
C      ECMT=COMENT
C
C      INCREMENT THE FLIGHT COUNTER
```



```
C
C 7  FLTSUM=FLTSUM+1
C
C  SORT THE ARRAY "TEST" IN ASCENDING ORDER.
C
C  DO 200  J2=1,N
C  DO 100  J1=1,N
C  IF (TEST(J2).GE.TEST(J1))GO TO 100
C  TEMP=TEST(J2)
C  TEST(J2)=TEST(J1)
C  TEST(J1)=TEMP
C  100 CONTINUE
C  200 CONTINUE
C
C  TEST TO SEE IF THE SIGHTING KEY HAS BEEN STORED.
C
C  IF(SUM.EQ.1) GO TO 9
C
C  READ A SIGHTING RECORD
C
C  8  READ(11,21,END=13) LOCK,BLK,SPECIE,DNITE,CMNT,STYPE,NOSC,TOTAL,
C  *IND, TABLE
C
C  INCREMENT THE SIGHTING RECORD COUNTER
C
C  SREC=SREC+1
C  SUM=1
C
C  STORE THE SIGHTING RECORD KEY
C
C  SKEY=LOCK
C  SDN=DNITE
C  SCMT=CMNT
C
C  TEST THE SIGHTING KEY
C
C  9  IF(SKEY.GT.EKEY) GO TO 10
C  IF(SKEY.NE.EKEY) WRITE(6,27) SREC,SKEY,BLK,SPECIE,SDN,CMNT,STYPE,
C  *NOSC,TOTAL,IND, TABLE
C  IF(SKEY.NE.EKEY) NOSTER=1
C
C  TEST THE DAY/NIGHT CODE FOR THE SIGHTING RECORD TO SEE IF IT
C  MATCHES THE DAY/NIGHT CODE ON THE EFFORT RECORD.
C
C  IF(SDN.NE.EDN) WRITE(6,26) SREC,LOCK
C  IF(SDN.NE.EDN) NOSTER=1
C
C  TEST THE BLOCK NUMBER IN WHICH THE SIGHTING TOOK PLACE.
C
C  CALL SEARCH
C  IF(MATCH.NE.1) WRITE(6,23) SREC,BLK,LOCK
C  IF(MATCH.NE.1) NOSTER=1
C
C  BRANCH BACK AND READ ANOTHER SIGHTING RECORD.
```


GO TO 8

CLEAR THE ARRAY "TEST".

10 DO 11 I=1,N

11 TEST(I)=0

N=0

GO TO 1

WRITE THE TOTAL NUMBER OF EFFORT AND SIGHTING RECORDS.

13 WRITE(6,24) RECORD,SREC

WRITE THE TOTAL NUMBER OF FLIGHTS MADE IN THIS MONTH.

WRITE(6,25) FLTSUM

IF(NOEFER.GT.0) WRITE(6,30)

IF(NOSTER.GT.0) WRITE(6,31)

WRITE(6,29)

***** FORMAT STATEMENTS *****

20 FORMAT(A11,I1,2I7,I3,I1,I3,(NUMBLK)I6)

21 FORMAT(A11,2X,I6,I4,2I1,I2,I4,I9,I4,I1)

22 FORMAT(1H0,' THE DANITE CODE ON THIS EFFORT RECORD, = ',I5,' KEY ='
*,A11,' DOES NOT MATCH THE DANITE CODES ON THE PREVIOUS EFFORT'
*, ' RECORDS')

23 FORMAT(1H0,' THE BLOCK NUMBER FOR THIS SIGHTING RECORD WAS NOT FOUN
*D ON THE FLIGHT PATH. SIGHTING RECORD =',I5,' BLOCK = ',I6,' KEY
*= ',A11)

24 FORMAT(1H1,' THE TOTAL NUMBER OF EFFORT RECORDS READ =',I7, //
*' THE TOTAL NUMBER OF SIGHTING RECORDS READ =',I5)

25 FORMAT(1H0,' THE TOTAL NUMBER OF FLIGHTS FOR THIS MONTH =',I4)

26 FORMAT(1H0,' THE DANITE CODE ON THIS SIGHTING RECORD = ',I5,' DOES
* NOT MATCH THE DANITE CODES ON THE PREVIOUS SIGHTING RECORDS.KEY
*= ',A11)

27 FORMAT(1H0,' THIS SIGHTING RECORD, NUMBER =',I5,' HAS NO MATCHING
* EFFORT RECORD',/,1H ,A11,I6,I4,2I1,I2,I4,I9,I4,I1)

29 FORMAT(1H1,'***** THE MERGE VERIFICATION PROGRAM HAS EXECUTED SUC
*CESSFULLY *****')

30 FORMAT(1H1,'***** POSSIBLE ERRORS WERE DETECTED ON THE EFFORT DAT
*A RECORDS *****')

31 FORMAT(1H0,'/'***** POSSIBLE ERRORS WERE DETECTED ON THE SIGHTING
* DATA RECORDS *****')
END

C SEARCH: A SUBROUTINE

SUBROUTINE SEARCH

C SEARCHES THE FLIGHT PATH TO MAKE SURE THE SIGHTING TOOK PLACE
C IN A BLOCK THAT IS IN IT

COMMON/ONE/ TEST(160),N,BLK,MATCH


```
INTEGER TEST, X, Y, HIGH, BLK
LOW=0
HIGH=N+1
MATCH=1
2  IF((HIGH-LOW).LT.2) GO TO 3
    X=(HIGH-LOW)/2
    Y=LOW+X
    IF(BLK.EQ.TEST(Y)) GO TO 4
    IF(BLK.GT.TEST(Y)) LOW=Y
    IF(BLK.LT.TEST(Y)) HIGH=Y
    GO TO 2
3  MATCH=2
4  RETURN
END
```


EFFORT RECORD
AS STORED IN THE DATABASE

Stored as a Variable Length Record

<u>Column</u>	<u>Contents</u>
1-2	Year
3-4	Month
5-6	Day
7-10	Pilot
11	Flight
12	Day/Night Code
13-19	Tachometer Time Out
20-26	Tachometer Time In
27-29	Plane
30	Comments
31-33	Number of Blocks Covered on that Flight
34-39	Block 1
40-45	Block 2
.	.
.	.
.	.
988-993	Block 160 (Maximum)

Effort Records That Are Over 77 Blocks Long
(greater than 500 characters long)

FLIGHT KEY	NUMBER OF BLOCKS
73080700071	95
73081700071	81
73081900071	106
73082100071	93
73090100071	129
74082900021	79
74090500021	93
76100500071	134
77082300091	79
82042200182	87
82072800181	84
83013100161	83

DATATRIEVE

DATATRIEVE is a database management software package that is useful for data editing, report writing and data summaries.

Shown here are the AMRMS effort and sighting record definitions used by DATATRIEVE. Also given is the list of domains. There is one domain for each effort and one domain for each sight file (2 domains per year).

Sometimes data editing can only be done using an indexed file. DTR allows ease of converting from AMRMS sequential files to indexed files. When the data is edited, reverse the process to obtain a sequential file.

To convert indexed to sequential:

```
READY T83 AS T83I
SHOW READY
DEFINE FILE T83
READY T83 AS T83S WRITE
SHOW READY
T83S=T83I
```


. DTP
V04Y-11 Datatrieve V1.3
DEC Query and Report System
Type HELP for help
DTR> SHOW ALL

Domains:

E62	E63	E64	E65
E66	E67	E68	E69
E70	E71	E72	E73
E74	E75	E76	E77
E78	E79	E80	E81
E82	E83	S62	S63
S64	S65	S66	S67
S68	S69	S70	S71
S72	S73	S74	S75
S76	S77	S78	S79
S80	S81	S82	S83
T83			

Records:

EFFORT SIGHT

The default directory is CDD\$TOP.NM42

No established collections.

No ready domains.

No loaded tables.

DTR> SHOW EFFORT

RECORD EFFORT

USING

ALLOCATION IS LEFT RIGHT

01 EFFORT.

03 FLIGHTKEY PIC X(11)

QUERY_NAME IS FK.

03 FLIGHT_KEY REDEFINES FLIGHTKEY

QUERY_NAME IS F_K.

06 DATE PIC X(6)

QUERY_NAME IS DA

EDIT_STRING IS XX-XX-XX.

06 YMD REDEFINES DATE.

09 YEAR PIC 9(2)

QUERY_NAME IS Y.

09 MONTH PIC 9(2)

QUERY_NAME IS M

EDIT_STRING IS Z9.

09 DAY PIC 9(2)

QUERY_NAME IS D

EDIT_STRING IS Z9.

06 PILOT PIC 9(4)

QUERY_NAME IS PI

EDIT_STRING IS ZZZ9.

06 FLIGHT PIC 9

QUERY_NAME IS FL.

03 DAY_NIGHT PIC 9

QUERY_NAME IS DN.

03 TIME_OUT PIC 9(6)V9

QUERY_NAME IS TO

EDIT_STRING IS ZZZZZ9.9.

03 TIME_IN PIC 9(6)V9

QUERY_NAME IS TI

EDIT_STRING IS ZZZZZ9.9.

03 PLANE PIC 9(3)

QUERY_NAME IS PL

EDIT_STRING IS ZZ9.

03 COMMENTS PIC X

QUERY_NAME IS C.

03 NUMBER_OF_BLOCKS PIC 9(3)

QUERY_NAME IS NB

EDIT_STRING IS ZZ9.

03 BLOCKS PIC 9(6)

OCCURS 0 TO 160 TIMES DEPENDING ON NUMBER_OF_BLOCKS

QUERY_NAME IS B

EDIT_STRING IS ZZZ999.

:

DTR> SHOW SIGHT

RECORD SIGHT

USING

ALLOCATION IS LEFT_RIGHT

01 SIGHT.

03 FLIGHTKEY PIC X(11)

QUERY_NAME IS FK.

03 FLIGHT_KEY REDEFINES FLIGHTKEY

QUERY_NAME IS F_K.

06 DATE PIC X(6)

QUERY_NAME IS DA

EDIT_STRING IS XX-XX-XX.

06 YMD REDEFINES DATE.

09 YEAR PIC 9(2)

QUERY_NAME IS Y.

09 MONTH PIC 9(2)

QUERY_NAME IS M

EDIT_STRING IS Z9.

09 DAY PIC 9(2)

QUERY_NAME IS D

EDIT_STRING IS Z9.

06 PILOT PIC 9(4)

QUERY_NAME IS PI

EDIT_STRING IS ZZZ9.

06 FLIGHT PIC 9

QUERY_NAME IS FL.

03 FILLER PIC X(2).

03 BLOCK PIC 9(6)

QUERY_NAME IS B

EDIT_STRING IS ZZZ999.

03 SPECIES PIC 9(4)

QUERY_NAME IS SP

EDIT_STRING IS ZZZ9.

03 DAY_NIGHT PIC 9

QUERY_NAME IS DN.

03 COMMENTS PIC X

QUERY_NAME IS C.

03 SIGHT_TYPE PIC 9(2)

QUERY_NAME IS ST

EDIT_STRING IS Z9.

03 NUMBER_OF_SCHOOLS PIC 9(4)

QUERY_NAME IS NS

EDIT_STRING IS ZZZ9.

03 TOTAL_TONNAGE PIC 9(9)

QUERY_NAME IS TT

EDIT_STRING IS ZZZZZZZZ9.

03 NUMBER_OF_INDIVIDUALS PIC 9(4)

QUERY_NAME IS NI

EDIT_STRING IS ZZZ9.

03 SPECIES_CODE_TABLE PIC X

QUERY_NAME IS SCT.

:

DTR> SHOW S83

DOMAIN S83 USING SIGHT ON UDISK6:[NM42.SIGHT]S83.D:

DTR> SHOW E83

DOMAIN E83 USING EFFORT ON UDISK6:[NM42.EFFORT]E83.D:

TAPES

On VAX Account NM42

Tapes

AERIAL 9/1600

AMRMS 9/1600

Both tapes contain the same information.

Savesets on the tapes:

EDIT.BCK
EFFORT.BCK
OLD.BCK
SIGHT.BCK
FMS.BCK

The savesets correspond to the disk subdirectories.

Data on tape is complete through 1982, with part of 1983.

Handy commands:

1. To read from tape to disk
\$OPINFO <tape> 1600
\$MOUNT/FOREIGN MTA0:
\$BACKUP/LOG MTA0:<saveset> <target directory>
2. To get a list of savesets in order written
\$MOUNT MTA0: <volume> <internal label>
\$DIR MTA0:
3. To write from disk to tape
 - A. First saveset on a tape (initialize tape)
\$BACKUP/LOG <source directory> MTA0:<saveset>
 - B. Subsequent savesets
\$BACKUP/LOG/NOREWIND <source directory> MTA0:<saveset>
4. To get a saveset listing
\$BACKUP/LOG/LIST=<filename> MTA0:<saveset>

Listing of save set

Save set: EDIT.BCK
 Written by: NM42
 UIC: [075.053]
 Date: 21-JUN-1983 10:18:14.57
 Command: BACKUP/LOG/LABEL=AERIAL/LIST=EDIT.BCK [NM42.EDIT] XTA0:EDIT.BCK
 Operating system: VAX/VMS version V3.1
 BACKUP version: V3.1
 CPU ID register: 013001E2
 Node name: _RV1::
 Written on: _XTA0:
 Block size: 8464
 Group size: 10
 Buffer count: 3

[000000]000000.DIR;1	10 20-MAY-1982 21:55
[000000]NM42.DIR;1	5 24-JUL-1982 11:28
[NM42]EDIT.DIR;1	3 26-AUG-1982 16:00
[NM42.EDIT]BLOCK.DAT;1	11 3-AUG-1982 10:47
[NM42.EDIT]EDISK.COM;1	1 11-MAY-1983 09:36
[NM42.EDIT]EDISK.EXE;2	8 13-APR-1983 16:53
[NM42.EDIT]EDISK.FOR;1	10 13-APR-1983 16:48
[NM42.EDIT]EEDIT.COM;1	1 11-MAY-1983 09:36
[NM42.EDIT]EEDIT.EXE;5	12 14-APR-1983 14:13
[NM42.EDIT]EEDIT.FOR;1	27 14-APR-1983 14:12
[NM42.EDIT]FILPRIN.COM;1	1 4-MAY-1983 10:41
[NM42.EDIT]FMSDISK.COM;1	1 11-MAY-1983 09:37
[NM42.EDIT]FMSDISK.EXE;7	7 11-MAY-1983 10:18
[NM42.EDIT]FMSDISK.FOR;1	9 11-MAY-1983 10:16
[NM42.EDIT]INDEX.EXE;1	10 26-JAN-1983 16:45
[NM42.EDIT]INDEX.FOR;2	9 26-JAN-1983 16:42
[NM42.EDIT]MERGE.COM;1	1 11-MAY-1983 09:38
[NM42.EDIT]MERGE.EXE;1	10 13-APR-1983 14:42
[NM42.EDIT]MERGE.FOR;1	22 29-OCT-1982 15:52
[NM42.EDIT]ORB.DAT;1	16 9-SEP-1982 15:26
[NM42.EDIT]RECCNT.EXE;1	6 18-APR-1983 10:06
[NM42.EDIT]RECCNT.FOR;1	3 29-NOV-1982 14:09
[NM42.EDIT]SDISK.COM;1	1 11-MAY-1983 09:38
[NM42.EDIT]SDISK.EXE;1	7 13-APR-1983 14:25
[NM42.EDIT]SDISK.FOR;1	9 29-OCT-1982 11:11
[NM42.EDIT]SEEDIT.COM;1	1 11-MAY-1983 09:38
[NM42.EDIT]SEEDIT.EXE;3	11 4-MAY-1983 10:45
[NM42.EDIT]SEEDIT.FOR;1	30 4-MAY-1983 10:43
[NM42.EDIT]SORT.COM;1	4 13-MAY-1983 12:55
[NM42.EDIT]VTOF.FOR;1	3 27-OCT-1982 09:17

Total of 30 files, 249 blocks
 End of save set

Listing of save set

Save set: EFFORT.BCK
Written by: NM42
UIC: [075,053]
Date: 21-JUN-1983 10:21:01.63
Command: BACKUP/LOG/NOREWIND/LIST=EFFORT.BCK [NM42.EFFORT] XTA0:EFFORT.BCK
Operating system: VAX/VMS version V3.1
BACKUP version: V3.1
CPU ID register: 013001E2
Node name: _RV1::
Written on: _XTA0:
Block size: 8464
Group size: 10
Buffer count: 3

[000000]000000.DIR:1	10	20-MAY-1982 21:55
[000000]NM42.DIR:1	5	24-JUL-1982 11:28
[NM42]EFFORT.DIR:1	3	26-AUG-1982 16:02
[NM42.EFFORT]E62.D:2	14	15-OCT-1982 08:53
[NM42.EFFORT]E63.D:3	113	15-OCT-1982 08:55
[NM42.EFFORT]E64.D:3	138	15-OCT-1982 08:56
[NM42.EFFORT]E65.D:3	142	15-OCT-1982 08:56
[NM42.EFFORT]E66.D:3	179	15-OCT-1982 08:57
[NM42.EFFORT]E67.D:3	172	15-OCT-1982 08:58
[NM42.EFFORT]E68.D:3	178	15-OCT-1982 08:59
[NM42.EFFORT]E69.D:3	241	15-OCT-1982 09:00
[NM42.EFFORT]E70.D:3	98	15-OCT-1982 09:02
[NM42.EFFORT]E71.D:3	261	15-OCT-1982 09:02
[NM42.EFFORT]E72.D:3	181	15-OCT-1982 09:08
[NM42.EFFORT]E73.D:3	264	15-OCT-1982 09:09
[NM42.EFFORT]E74.D:3	183	15-OCT-1982 09:10
[NM42.EFFORT]E75.D:3	178	15-OCT-1982 09:10
[NM42.EFFORT]E76.D:3	278	15-OCT-1982 09:11
[NM42.EFFORT]E77.D:3	238	15-OCT-1982 09:13
[NM42.EFFORT]E78.D:3	166	15-OCT-1982 09:13
[NM42.EFFORT]E79.D:5	238	30-NOV-1982 15:17
[NM42.EFFORT]E80.D:2	109	15-OCT-1982 09:15
[NM42.EFFORT]E81.D:2	197	30-NOV-1982 15:53
[NM42.EFFORT]E82.D:1	228	4-MAY-1983 11:26
[NM42.EFFORT]E83.D:1	47	12-MAY-1983 10:13

Total of 25 files, 3861 blocks
End of save set

Listing of save set

Save set: FORMS.BCK
Written by: NM42
UIC: [075.053]
Date: 21-JUN-1983 10:26:25.77
Command: BACKUP/LOG/NOREWIND/LIST=FORMS.BCK [NM42.FORMS] XTA0:FORMS.BCK
Operating system: VAX/VMS version V3.1
BACKUP version: V3.1
CPU ID register: 013001E2
Node name: _RV1::
Written on: _XTA0:
Block size: 8464
Group size: 10
Buffer count: 3

[000000]000000.DIR;1	10	20-MAY-1982	21:55
[000000]NM42.DIR;1	5	24-JUL-1982	11:28
[NM42]FORMS.DIR;1	1	7-APR-1983	14:30
[NM42.FORMS]AMRMS.FLB;1	4	8-MAR-1983	16:32
[NM42.FORMS]CODE.EXE;3	47	18-MAY-1983	16:23
[NM42.FORMS]CODE.FOR;1	11	18-MAY-1983	16:21
[NM42.FORMS]EFFORT.FRM;1	1	8-MAR-1983	16:31
[NM42.FORMS]SIGHT.FRM;1	2	7-MAR-1983	14:25

Total of 8 files, 81 blocks
End of save set

Listing of save set

Save set: OLD.BCK
 Written by: NM42
 UIC: [075,253]
 Date: 21-JUN-1983 10:29:04.14
 Command: BACKUP/LOG/NOREWIND/LIST=OLD.BCK [NM42.OLD] XTA0:OLD.BCK
 Operating system: VAX/VMS version V3.1
 BACKUP version: V3.1
 CPU ID register: 013001E2
 Node name: _RV1::
 Written on: _XTA0:
 Block size: 8464
 Group size: 10
 Buffer count: 3

[000000]000000.DIR:1	10	20-MAY-1982 21:55
[000000]NM42.DIR:1	5	24-JUL-1982 11:28
[NM42]OLD.DIR:1	4	18-OCT-1982 16:25
[NM42.OLD]BLOCK.DAT:1	11	3-AUG-1982 10:47
[NM42.OLD]E.FOR:1	3	30-SEP-1982 15:30
[NM42.OLD]EDISK.COM:1	1	26-AUG-1982 16:26
[NM42.OLD]EDISK.FOR:1	7	23-SEP-1982 13:37
[NM42.OLD]EEDIT.COM:1	1	2-SEP-1982 15:12
[NM42.OLD]EEDIT.FOR:1	35	2-SEP-1982 15:18
[NM42.OLD]EFFORT62.DAT:1	22	27-AUG-1982 10:18
[NM42.OLD]EFFORT63.DAT:1	169	28-SEP-1982 16:16
[NM42.OLD]EFFORT64.DAT:1	212	29-SEP-1982 09:31
[NM42.OLD]EFFORT65.DAT:1	219	16-SEP-1982 15:13
[NM42.OLD]EFFORT66.DAT:1	275	30-AUG-1982 11:03
[NM42.OLD]EFFORT67.DAT:1	268	28-SEP-1982 13:24
[NM42.OLD]EFFORT68.DAT:1	274	12-OCT-1982 11:39
[NM42.OLD]EFFORT69.DAT:1	373	23-SEP-1982 14:25
[NM42.OLD]EFFORT70.DAT:1	155	16-SEP-1982 15:25
[NM42.OLD]EFFORT71.DAT:1	405	30-AUG-1982 11:07
[NM42.OLD]EFFORT72.DAT:1	281	23-SEP-1982 14:33
[NM42.OLD]EFFORT73.DAT:1	412	23-SEP-1982 15:06
[NM42.OLD]EFFORT74.DAT:1	283	7-SEP-1982 15:08
[NM42.OLD]EFFORT75.DAT:1	275	30-AUG-1982 11:14
[NM42.OLD]EFFORT76.DAT:1	435	27-SEP-1982 11:12
[NM42.OLD]EFFORT77.DAT:1	371	30-AUG-1982 11:17
[NM42.OLD]EFFORT78.DAT:1	258	8-SEP-1982 10:45
[NM42.OLD]EFFORT79.DAT:2	377	16-NOV-1982 13:39
[NM42.OLD]EFFORT80.DAT:1	169	8-SEP-1982 10:51
[NM42.OLD]EFFORT81.DAT:2	307	17-NOV-1982 15:17
[NM42.OLD]EFFORT82.DAT:1	57	16-NOV-1982 14:07
[NM42.OLD]ESORT.COM:1	1	27-AUG-1982 13:23
[NM42.OLD]EZERO.FOR:1	2	15-OCT-1982 14:04
[NM42.OLD]INDEX.FOR:1	7	29-JUL-1982 14:56
[NM42.OLD]MERGE.COM:1	1	29-SEP-1982 10:39
[NM42.OLD]MERGE.FOR:1	27	29-SEP-1982 11:36
[NM42.OLD]ORB.DAT:1	16	9-SEP-1982 15:26
[NM42.OLD]S.FOR:1	10	16-SEP-1982 12:03
[NM42.OLD]SDISK.COM:1	1	26-AUG-1982 16:27
[NM42.OLD]SDISK.FOR:1	9	23-SEP-1982 13:42
[NM42.OLD]SEEDIT.COM:1	1	2-SEP-1982 15:11
[NM42.OLD]SEEDIT.FOR:1	47	2-SEP-1982 15:14
[NM42.OLD]SIGHT62.DAT:2	67	30-SEP-1982 15:14
[NM42.OLD]SIGHT63.DAT:1	578	4-OCT-1982 15:55
[NM42.OLD]SIGHT64.DAT:1	452	3-SEP-1982 08:41

[NM42.OLD]SIGHT65.DAT;1	330	4-OCT-1982	16:08
[NM42.OLD]SIGHT66.DAT;2	397	4-OCT-1982	16:11
[NM42.OLD]SIGHT67.DAT;1	429	4-OCT-1982	16:35
[NM42.OLD]SIGHT68.DAT;1	340	4-OCT-1982	16:46
[NM42.OLD]SIGHT69.DAT;1	426	7-SEP-1982	14:51
[NM42.OLD]SIGHT70.DAT;1	210	13-SEP-1982	14:54
[NM42.OLD]SIGHT71.DAT;1	440	4-OCT-1982	16:52
[NM42.OLD]SIGHT72.DAT;1	274	3-SEP-1982	10:27
[NM42.OLD]SIGHT73.DAT;1	552	3-SEP-1982	10:37
[NM42.OLD]SIGHT74.DAT;1	380	4-OCT-1982	16:55
[NM42.OLD]SIGHT75.DAT;1	270	13-SEP-1982	15:01
[NM42.OLD]SIGHT76.DAT;1	308	5-OCT-1982	11:45
[NM42.OLD]SIGHT77.DAT;1	336	3-SEP-1982	11:04
[NM42.OLD]SIGHT78.DAT;1	260	4-OCT-1982	16:57
[NM42.OLD]SIGHT79.DAT;2	339	16-NOV-1982	13:37
[NM42.OLD]SIGHT80.DAT;1	246	4-OCT-1982	17:00
[NM42.OLD]SIGHT81.DAT;2	376	17-NOV-1982	15:21
[NM42.OLD]SIGHT82.DAT;1	79	16-NOV-1982	14:09
[NM42.OLD]SSORT.COM;1	1	27-AUG-1982	13:24
[NM42.OLD]SZERO.FOR;1	3	15-OCT-1982	09:48

Total of 64 files, 12889 blocks
End of save set

Listing of save set

Save set: SIGHT.BCK
Written by: NM42
UIC: [075,053]
Date: 21-JUN-1983 10:39:05.89
Command: BACKUP/LOG/NOREWIND/LIST=SIGHT.BCK [NM42.SIGHT] XTA0:SIGHT.BCK
Operating system: VAX/VMS version V3.1
BACKUP version: V3.1
CPU ID register: 013001E2
Node name: _RV1::
Written on: _XTA0:
Block size: 8464
Group size: 10
Buffer count: 3

[000000]000000.DIR;1	10	20-MAY-1982	21:55
[000000]NM42.DIR;1	5	24-JUL-1982	11:28
[NM42]SIGHT.DIR;1	3	26-AUG-1982	16:01
[NM42.SIGHT]S62.D;2	40	15-OCT-1982	09:52
[NM42.SIGHT]S63.D;2	350	1-DEC-1982	12:43
[NM42.SIGHT]S64.D;2	273	15-OCT-1982	12:40
[NM42.SIGHT]S65.D;3	197	1-DEC-1982	12:45
[NM42.SIGHT]S66.D;2	241	15-OCT-1982	12:46
[NM42.SIGHT]S67.D;2	258	15-OCT-1982	10:02
[NM42.SIGHT]S68.D;2	203	15-OCT-1982	12:50
[NM42.SIGHT]S69.D;2	255	15-OCT-1982	12:51
[NM42.SIGHT]S70.D;2	126	15-OCT-1982	10:05
[NM42.SIGHT]S71.D;3	266	1-DEC-1982	12:47
[NM42.SIGHT]S72.D;2	165	15-OCT-1982	13:15
[NM42.SIGHT]S73.D;2	329	15-OCT-1982	13:16
[NM42.SIGHT]S74.D;2	226	15-OCT-1982	13:20
[NM42.SIGHT]S75.D;2	160	15-OCT-1982	13:27
[NM42.SIGHT]S76.D;2	185	15-OCT-1982	13:37
[NM42.SIGHT]S77.D;2	202	15-OCT-1982	10:31
[NM42.SIGHT]S78.D;2	154	1-DEC-1982	12:49
[NM42.SIGHT]S79.D;5	201	30-NOV-1982	15:15
[NM42.SIGHT]S80.D;2	145	1-DEC-1982	12:51
[NM42.SIGHT]S81.D;4	223	1-DEC-1982	12:56
[NM42.SIGHT]S82.D;1	206	4-MAY-1983	11:24
[NM42.SIGHT]S83.D;1	43	11-MAY-1983	10:19

Total of 25 files, 4466 blocks
End of save set

USEFUL REFERENCES FOR DATA EDITING

SUBJECT	REFERENCE
Form Management System	VAX-11 FMS Software Reference Manual
SOS Editor	VAX-11 Text Editing Reference Manual
Datatrieve	VAX-11 Datatrieve Reference Manual
EDT Editor	VAX EDT Editor Manual
Tape Management	VAX/Vms Command Language User's Guide

AERIAL MARINE RESOURCES MONITORING SYSTEM

YEARLY SUMMARIES

YEAR	NUMBER OF RECORDS		NUMBER
1962	-----		OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	-	-	-
FEB	-	-	-
MAR	-	-	-
APR	-	-	-
MAY	-	-	-
JUN	-	-	-
JUL	-	-	-
AUG	-	-	-
SEP	15	130	15
OCT	19	128	19
NOV	6	75	6
DEC	8	88	8
TOTAL	48	421	48

YEAR	NUMBER OF RECORDS		NUMBER
1963	-----		OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	32	304	32
FEB	24	264	24
MAR	23	190	23
APR	22	169	22
MAY	21	184	21
JUN	16	105	16
JUL	14	146	14
AUG	69	530	69
SEP	64	644	64
OCT	63	468	63
NOV	67	546	67
DEC	24	181	24
TOTAL	439	3731	439

YEAR	NUMBER OF RECORDS		NUMBER
1964	-----		OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	11	105	11
FEB	22	145	22
MAR	55	373	55
APR	34	186	34
MAY	4	16	4
JUN	10	31	10
JUL	24	176	24
AUG	112	785	112
SEP	99	600	99
OCT	35	168	35
NOV	55	235	55
DEC	23	83	23
-----	-----	-----	-----
TOTAL	484	2903	484

YEAR	NUMBER OF RECORDS		NUMBER
1965	-----		OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	16	103	16
FEB	15	94	15
MAR	40	262	40
APR	17	66	17
MAY	22	124	22
JUN	21	147	21
JUL	34	156	34
AUG	81	437	81
SEP	78	348	78
OCT	60	249	60
NOV	22	48	22
DEC	24	66	24
-----	-----	-----	-----
TOTAL	430	2100	430

YEAR 1966	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	29	89	29
FEB	31	102	31
MAR	54	168	54
APR	56	236	56
MAY	16	89	16
JUN	31	109	31
JUL	90	432	90
AUG	108	555	108
SEP	39	161	39
OCT	61	324	61
NOV	57	188	57
DEC	25	111	25
TOTAL	597	2564	597

YEAR 1967	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	47	335	47
FEB	45	321	45
MAR	63	336	63
APR	36	129	36
MAY	34	128	34
JUN	35	176	35
JUL	66	359	66
AUG	95	450	95
SEP	49	234	49
OCT	26	105	26
NOV	34	96	34
DEC	23	81	23
TOTAL	553	2750	553

YEAR 1968	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	66	320	66
FEB	47	173	47
MAR	48	156	48
APR	12	44	12
MAY	19	44	19
JUN	25	56	25
JUL	53	237	53
AUG	98	480	98
SEP	93	359	93
OCT	26	60	26
NOV	31	103	31
DEC	31	125	31
TOTAL	549	2157	549

YEAR 1969	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	30	132	30
FEB	38	122	38
MAR	50	190	50
APR	74	335	74
MAY	33	114	33
JUN	46	191	46
JUL	99	336	99
AUG	115	384	115
SEP	83	257	83
OCT	83	230	83
NOV	72	188	72
DEC	71	241	71
TOTAL	794	2720	794

YEAR 1970	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	59	253	59
FEB	63	256	63
MAR	91	435	91
APR	29	104	29
MAY	37	145	37
JUN	24	134	24
JUL	4	11	4
AUG	-	-	-
SEP	-	-	-
OCT	-	-	-
NOV	-	-	-
DEC	-	-	-
TOTAL	307	1338	307

YEAR 1971	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	61	281	61
FEB	37	129	37
MAR	43	157	43
APR	39	106	39
MAY	41	108	41
JUN	46	223	46
JUL	61	191	61
AUG	106	447	106
SEP	148	592	148
OCT	81	251	81
NOV	52	186	52
DEC	56	159	56
TOTAL	771	2830	771

YEAR 1972	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	42	135	42
FEB	36	129	36
MAR	39	165	39
APR	49	144	49
MAY	41	108	41
JUN	33	166	33
JUL	21	84	21
AUG	65	214	65
SEP	95	300	95
OCT	65	107	65
NOV	50	111	50
DEC	41	94	41
TOTAL	577	1757	577

YEAR 1973	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	72	225	72
FEB	14	47	14
MAR	17	84	17
APR	70	172	70
MAY	46	142	46
JUN	13	50	13
JUL	72	357	72
AUG	115	640	115
SEP	97	391	97
OCT	122	584	122
NOV	98	445	98
DEC	80	367	80
TOTAL	816	3504	816

YEAR 1974	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	30	90	30
FEB	50	169	50
MAR	9	50	9
APR	14	28	14
MAY	3	26	3
JUN	6	27	6
JUL	69	407	69
AUG	105	513	105
SEP	48	252	48
OCT	61	225	61
NOV	85	356	85
DEC	67	263	67
TOTAL	547	2406	547

YEAR 1975	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	71	309	71
FEB	46	168	46
MAR	34	82	34
APR	69	159	69
MAY	46	119	46
JUN	-	-	-
JUL	62	155	62
AUG	63	137	63
SEP	57	126	57
OCT	60	161	60
NOV	69	153	69
DEC	52	136	52
TOTAL	629	1705	629

YEAR 1976	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	78	189	78
FEB	33	51	33
MAR	83	215	83
APR	60	168	60
MAY	32	92	32
JUN	16	35	16
JUL	87	222	87
AUG	91	236	91
SEP	112	244	112
OCT	64	133	64
NOV	87	200	87
DEC	96	180	96
TOTAL	839	1965	839

YEAR 1977	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	42	111	42
FEB	20	75	20
MAR	77	231	77
APR	66	164	66
MAY	41	124	41
JUN	57	199	57
JUL	58	193	58
AUG	72	212	72
SEP	77	185	77
OCT	75	217	75
NOV	93	276	93
DEC	51	160	51
TOTAL	729	2147	729

YEAR	NUMBER OF RECORDS		NUMBER
1978	-----		OF
MONTH	EFFORT	SIGHTING	FLIGHTS

JAN	25	52	25
FEB	22	68	22
MAR	46	120	46
APR	36	87	36
MAY	-	-	-
JUN	3	15	3
JUL	44	110	44
AUG	59	186	59
SEP	77	226	77
OCT	89	308	89
NOV	76	282	76
DEC	54	182	54

TOTAL	531	1636	531

YEAR	NUMBER OF RECORDS		NUMBER
1979	-----		OF
MONTH	EFFORT	SIGHTING	FLIGHTS

JAN	50	22	50
FEB	20	79	20
MAR	45	46	45
APR	111	314	111
MAY	117	411	117
JUN	67	203	67
JUL	83	269	83
AUG	97	263	97
SEP	83	183	83
OCT	51	126	51
NOV	41	149	41
DEC	24	72	24

TOTAL	789	2137	789

YEAR 1980	NUMBER OF RECORDS		NUMBER OF FLIGHTS
MONTH	EFFORT	SIGHTING	
JAN	23	104	23
FEB	24	109	24
MAR	27	110	27
APR	27	96	27
MAY	25	64	25
JUN	38	166	38
JUL	24	77	24
AUG	38	169	38
SEP	34	116	34
OCT	40	190	40
NOV	43	193	43
DEC	33	144	33
TOTAL	376	1538	376

YEAR 1981	NUMBER OF RECORDS		NUMBER OF FLIGHTS
MONTH	EFFORT	SIGHTING	
JAN	32	158	32
FEB	30	153	30
MAR	40	164	40
APR	45	196	45
MAY	43	150	43
JUN	48	153	48
JUL	84	341	84
AUG	80	297	80
SEP	74	253	74
OCT	60	195	60
NOV	56	174	56
DEC	37	140	37
TOTAL	629	2374	629

YEAR 1982	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	53	250	53
FEB	37	130	37
MAR	47	178	47
APR	19	71	19
MAY	71	240	71
JUN	71	179	71
JUL	93	332	93
AUG	86	239	86
SEP	69	234	69
OCT	52	164	52
NOV	28	164	28
DEC	40	101	40
TOTAL	666	2196	666

YEAR 1983	NUMBER OF RECORDS		NUMBER OF
MONTH	EFFORT	SIGHTING	FLIGHTS
JAN	61	224	61
FEB	38	152	38
MAR	14	73	14
APR	-	-	-
MAY	-	-	-
JUN	-	-	-
JUL	-	-	-
AUG	-	-	-
SEP	-	-	-
OCT	-	-	-
NOV	-	-	-
DEC	-	-	-
TOTAL	113	449	113

RECCNT

Record Counting Program

Purpose

RECCNT, the record counting program, will give a monthly summary of the number of sighting and effort records.

Input

The input is the year for which the record count summary is desired. Both the sighting and effort files must be completely edited, and both must be in the same subdirectory. The end result will be a list of record counts by month for effort and sighting, as well as the yearly total.

To use this program, get in the subdirectory where the files to be counted are, and type RUN [NM42.EDIT]RECCNT.


```
CHARACTER*2 YEAR
CHARACTER*5 E,S
INTEGER EM(12),SM(12),M,ET,ST
PRINT*, 'ENTER DESIRED YEAR'
READ(5,1)YEAR
1 FORMAT(A)
E='E'//YEAR//'.D'
S='S'//YEAR//'.D'
DATA EM/12*0/,SM/12*0/,ET/0/,ST/0/
C OPEN THE FILES
OPEN(UNIT=10,TYPE='OLD',NAME=E,READONLY)
OPEN(UNIT=20,TYPE='OLD',NAME=S,READONLY)
C READ THE MONTH ON THE EFFORT RECORD
5 READ(10,15,END=25)M
15 FORMAT(2X,I2)
C INCREMENT THE COUNTER FOR THAT MONTH
EM(M)=EM(M)+1
C INCREMENT THE TOTAL EFFORT RECORD COUNTER
ET=ET+1
C RETURN FOR NEXT EFFORT RECORD
GO TO 5
C READ THE MONTH ON THE SIGHT RECORD
25 READ(20,15,END=35)M
C INCREMENT COUNTER FOR THAT MONTH
SM(M)=SM(M)+1
C INCREMENT TOTAL SIGHT RECORD COUNTER
ST=ST+1
C RETURN FOR A NEW SIGHT RECORD
GO TO 25
C PRINT THE RESULTS - THE NUMBER OF RECORDS PER
C MONTH IN THE EFFORT AND SIGHT FILES
35 DO 100 I=1,12
WRITE(6,45)EM(I),SM(I)
45 FORMAT(2X,I7,3X,I7)
100 CONTINUE
WRITE(6,45)ET,ST
STOP
END
```


CHANGE OF FORMATS

In 1982, the AMRMS record formats were modified. It was no longer necessary to have files composed of card-image records. For a complete description of the old record formats and coding, refer to Caruso (1979). The new effort records are variable length, one record per flight, which eliminates the need to repeat header information. The sighting records were compacted, and the sighting type codes were redefined slightly. A new field for individuals was added, because formerly this information was either lost, or, if present, was difficult to retrieve. The basic principals of coding have remained the same.

VTOF

Variable to Fixed Length Effort Records Program

Purpose

The program VTOF takes a file of variable length effort records and converts it to a file of fixed-length records. This program is to be used if the AMRMS data must be moved to a computer that does not support variable-length records. The effort records will be in the same format as the effort coding sheet.

Input

The input is a file of variable-length effort records, with one record per flight.

A file of fixed-length effort records is produced. There will be one to several records per flight in the output file.


```
C
C PROGRAM: VTOF.FOR
C
C THIS PROGRAM WILL CREATE A FIXED LENGTH RECORD FILE FROM
C THE VARIABLE LENGTH EFFORT FILE
C
C     CHARACTER*25 OLDFILE,NEWFILE
C     CHARACTER*30 DATA
C     INTEGER NB,B(160)
C
C GET FILE NAMES AND OPEN THE FILES
C
C     PRINT*,'OLD FILE?'
C     READ(5,5)OLDFILE
C 5  FORMAT(A)
C     OPEN(UNIT=10,NAME=OLDFILE,TYPE='OLD',FORM='FORMATTED',
C *RECORDTYPE='VARIABLE',RECL=993)
C     PRINT*,'NEW FILE?'
C     READ(5,5)NEWFILE
C     OPEN(UNIT=20,NAME=NEWFILE,TYPE='NEW',CARRIAGECONTROL='LIST')
C
C READ A RECORD
C
C 30  READ(10,15,END=99) DATA,NB,(B(I),I=1,NB)
C 15  FORMAT(A30,I3,(NB)I6)
C
C IF THE NUMBER OF BLOCKS IS NOT EVENLY DIVISIBLE BY 8,
C ADD BLOCKS OF 0 UNTIL THE TOTAL NUMBER IS A MULTIPLE OF 8.
C
C     DO 40 I=1,7
C     IF (MOD(NB,8).EQ.0) GO TO 50
C     NB=NB+1
C     B(NB)=0
C 40  CONTINUE
C
C WRITE THE RECORDS FOR THE FLIGHT. EACH RECORD CONTAINS THE
C FLIGHT INFORMATION, 8 BLOCKS, AND A SEQUENCE NUMBER SO
C THE FLIGHT CAN BE KEPT IN ORDER. THERE COULD BE MORE
C THAN ONE RECORD PER FLIGHT.
C
C 50  DO 60 J=1,NB/8
C     L=8*(J-1)
C     WRITE(20,25)DATA,(B(K),K=L+1,L+8),J
C 25  FORMAT(A30,8I6.6,I2.2)
C 60  CONTINUE
C
C RETURN TO FETCH ANOTHER RECORD. THIS WILL BEGIN ANOTHER FLIGHT.
C
C     GO TO 30
C 99  STOP
C     END
```


REFERENCES

- Caruso, Joseph P. 1979. Aerial Marine Resources Monitoring System. Data coding manual. Administrative Report LJ-79-9, 99pp. Southwest Fisheries Center, La Jolla Laboratory, NMFS, NOAA, La Jolla, CA 92038.
- MacCall, Alec. 1975. Investigation of aerial spotter variation in reported abundance of anchovies. Contrib. 19, Anchovy Meeting Workshop, July 21-22, 1975, Southwest Fisheries Center, La Jolla, California.
- Squire, James L., Jr. 1972. Apparent abundance of some pelagic marine fishes off the southern and central California coast as surveyed by an airborne monitoring program. Fish. Bull., U.S. 70(3):1005-1019.
- Squire, James L., Jr. 1983. Abundance of pelagic resources off California, 1963-78, as measured by an airborne fish monitoring program. NOAA Tech. Rep. NMFS SSRF-762, 75pp.

From 1963 to 1978, the index of apparent abundance for northern anchovy was calculated by hand. These calculations were time-consuming, so the calculation technique was put on the computer. The computer tries to follow closely the method used by Squire (1972), when he did the calculations by hand. This gives a step-by-step description of how the computer calculates the night index, and will suggest reasons for the discrepancy between the computer's index and Squire's index.

The Anchovy Night Index Calculation

- A. Count the total number of blocks searched.
 - 0. Each year has one data file containing all effort records for that year. One data record per flight.
 - 1. Zero the total block counter for the year.
 - 2. Read a record from the data file.
 - 3. Check the day/night code.
 - If night, given a weight of 1.
 - If the pilot did not indicate the time of day, the flight is assumed to be both day and night. This is given a weight of 0.5 because only the night is wanted.
 - 4. Count the blocks in the record.
 - Eliminate zeros, unknown areas and overland flight areas.
 - 5. Multiply the blocks counted on that record by the weight.
 - 6. Add the record block count to the total block count.
 - 7. Repeat steps 2-6 until all effort records have been processed.
- B. Count the anchovy night sightings.
 - 0. Each year has a file containing all the sighting information for that year. There is one record per sighting.
 - 1. Zero the tonnage class counters for that year.
 - 2. Zero the record tonnage counter.
 - 3. Read the record from the file.
 - 4. Check the day/night code.
 - If night, weight=1.
 - If day and night, because the pilot did not indicate the time of day, weight=0.5 because only the night half of the flight is wanted.
 - 5. Check the species to make sure it is anchovy, or anchovy in combination with another species.
 - If anchovy, weight=weight*1.
 - If anchovy + 1 other species, weight=weight*0.5.
 - If anchovy + 2 other species, weight=weight*0.33.
 - 6. Get the tonnage from the record.
 - 7. If the tonnage is zero, it indicates the pilot did not specify a tonnage, so tonnage is assigned according to the pilot's comment.

COMMENT	TONS
Yes	100
No	0
Scattered	50
Many	200
Few	100
Heavy	200
Small	50
Some	25

8. Add one to the proper tonnage class.

CLASS	RANGE (TONS)	TONNAGE RANGE VALUE (X)
1	0 - 400	2
2	400 - 1000	7
3	1000 - 10000	55
4	10000 - 20000	150

9. Repeat steps 2-8 until all records have been processed.

C. Calculating the anchovy night index.

1. The formula for the index is

$$\frac{\sum_{i=1}^4 N_i X_i}{N_t}$$

N_i is the number of block area flights where species occurred at value X_i
 X_i is the tonnage range value
 N_t is the total number of block area flights during the year

2. The N_i 's have been totalled by counting the anchovy sightings.
3. The X_i 's are given (Part B, Step 8).
4. N_t was totalled by counting effort blocks.
5. Plugging the values into the formula gives the anchovy night index for all areas.

Differences Between Squire's Method And The Computer Method

A. Symbols used:



Block



Fish School



Actual Flight Path

Adjusted Flight Path

Squire only counted those blocks actually flown through.



The computer-adjusted flight path includes blocks with sightings that were not actually flown through.



The result is a greater number of effort blocks for the computer. The total effect of this would be to lower the index value. How much lower would depend on the increase in total blocks.

- B. Another source of error arises from the pilot not indicating the time of day of the flight. This affects not only the number of blocks flown through, but also the tonnage of anchovies. The total effect on the index is not known, as Squire always made each flight either day or night.
- C. Combinations of species also cause problems. Sometimes the pilot gives two species names but only one tonnage for the combination of them. For calculating the index it is assumed that half the tonnage was one species, and half was the other species. This causes an error because the animals do not form mixed groups with each group contributing the same amount. It is hard to estimate what the effect of this is on the index, because Squire always divided the data by hand on these combined species.
- D. The computer calculates the index over the entire area. It does not do the calculation by zones. Squire calculated indexes for several zones, then averaged the indexes of all the zones. Squire's average index will not equal the total area index calculated by the computer, unless the same amount of effort is expended in each zone. Therefore, this further contributes to the discrepancy between the computer index and Squire's index.

REFERENCE:

- Squire, J.L. 1972. Apparent abundance of some pelagic marine fishes off the southern and central coast as surveyed by an airborne monitoring program. U.S. Dept. of Commerce, Fish. Bull. 70(3):1005-1019.


```
C
C PROGRAM: INDEX.FOR
C
C THIS PROGRAM SET UP FOR ANCHOVY INDEX ONLY
C
      INTEGER BLOCK,B(160)
      CHARACTER*25 EFFORT,SIGHT
      CHARACTER*2 YEAR
      INTEGER TONS,BLOCKS,X(4),N(4),V(4),SIGHTTYPE,DN
      INTEGER ZEROS,C,NS,SPECIES,COUNT(6),BL
      REAL INDEX
C
C INITIALIZE VARIABLES
C
      DATA N/4*0/
      DATA V/400,1000,10000,20000/
      DATA X/2,7,55,150/
      DATA COUNT/6*0/
C
C OPEN NECESSARY FILES
C
      PRINT*,'ENTER DESIRED YEAR'
      READ(5,5)YEAR
5    FORMAT(A)
      EFFORT=' [NM42.EFFORT]E'//YEAR//'.D'
      SIGHT=' [NM42.SIGHT]S'//YEAR//'.D'
      OPEN(UNIT=10,TYPE='OLD',NAME=EFFORT,FORM='FORMATTED',
      *RECORDTYPE='VARIABLE',RECL=993,READONLY)
      OPEN(UNIT=20,TYPE='OLD',NAME=SIGHT,READONLY)
C
C INITIALIZE MORE VARIABLES
C
      BL=0
      BLOCKS=0
      ZEROS=0
C
C COUNT NUMBER OF BLOCKS DURING DESIRED TIME OF DAY AND
C IN DESIRED LOCATION
C
      8    READ(10,10,END=99)DN,NUMBLK,(B(I),I=1,NUMBLK)
10    FORMAT(T12,I1,T31,I3,<NUMBLK>I6)
      W=1
      BL=0
      IF (DN.EQ.1) GO TO 8
      IF (DN.EQ.3) W=0.5
      DO 125 J=1,NUMBLK
      IF((B(J).NE.0).AND.(B(J).GT.999)) BL=BL+1
125   CONTINUE
      BL=BL*W
      BLOCKS=BLOCKS+BL
      GO TO 8
C
C COUNT THE SIGHTINGS
C
      99   READ(20,25,END=199)BLOCK,SPECIES,DN,C,SIGHTTYPE,NS,TONS
```



```
25  FORMAT(T14,I6,I4,2I1,I2,I4,I9)
      W=1
      IF (DN.EQ.1) GO TO 99
      IF (DN.EQ.3) W=0.5
      IF((SPECIES.NE.110).AND.(SPECIES.NE.1120).AND.
* (SPECIES.NE.1128).AND.
* (SPECIES.NE.1122).AND.(SPECIES.NE.1127).AND.(SPECIES.NE.1133)
* .AND.(SPECIES.NE.1126).AND.(SPECIES.NE.1135)) GO TO 99
      IF ((SPECIES.EQ.1120).OR.(SPECIES.EQ.1122).OR.(SPECIES.EQ.1127)
* .OR.(SPECIES.EQ.1128).OR.(SPECIES.EQ.1133)) W=W*0.5
      IF ((SPECIES.EQ.1126).OR.(SPECIES.EQ.1135)) W=W*0.33
      GO TO (100,200,300,400,500,600),SIGHTTYPE
```

C

C SCHOOLS

C

```
100  COUNT(1)=COUNT(1)+1
      CALL ADDEMUP(TONS,N,ZEROS,C,W)
      GO TO 99
```

C

C TONS

C

```
200  COUNT(2)=COUNT(2)+1
      CALL ADDEMUP(TONS,N,ZEROS,C,W)
      GO TO 99
```

C

C INDIVIDUALS

C

```
300  COUNT(3)=COUNT(3)+1
      GO TO 99
```

C

C SCHOOLS AND TOTAL TONNAGE

C

```
400  COUNT(4)=COUNT(4)+1
      CALL ADDEMUP(TONS,N,ZEROS,C,W)
      GO TO 99
```

C

C SCHOOLS AND INDIVIDUAL TONNAGE

C

```
500  COUNT(5)=COUNT(5)+1
      TONS=TONS/NS
      DO 56 J=1,NS
      CALL ADDEMUP(TONS,N,ZEROS,C,W)
56   CONTINUE
      GO TO 99
```

C

C NO DEFINITE INFORMATION

C

```
600  COUNT(6)=COUNT(6)+1
      CALL ADDEMUP(TONS,N,ZEROS,C,W)
      GO TO 99
```

C

C PRINT THE RESULTS

C

```
199  PRINT*, 'ANCHOVY YEARLY INDEX ', YEAR
      PRINT*, 'NUMBER OF EACH SIGHTING TYPE '
```



```
DO 140 I=1,6
140 PRINT*, I, ' ', COUNT(I)
PRINT*, 'ZEROS = ', ZEROS
WRITE(6,97) (N(I), I=1, 4)
97 FORMAT(4(3X, I9))
WRITE(6,97) (X(I), I=1, 4)
PRINT*, 'BLOCKS = ', BLOCKS
PRINT*, ' '
INDEX=0
DO 150 I=1, 4
150 INDEX=INDEX+N(I)*X(I)
PRINT*, 'SUMMATION = ', INDEX
PRINT*, ' '
INDEX=INDEX/BLOCKS
PRINT*, 'INDEX = ', INDEX
STOP
END

C
C
C ADDEMUP: A SUBROUTINE
C
C SUBROUTINE ADDEMUP(TONS, N, ZEROS, C, W)
C
C THIS ROUTINE TOTALS THE TONNAGE FOR THE FOUR TONNAGE RANGES.
C
C INTEGER TONS, N(4), V(4), ZEROS, C
C DATA V/400, 1000, 10000, 20000/
C
C IF THE TONNAGE IS 0, ASSIGN A TONNAGE ACCORDING TO THE COMMENT.
C
C IF (TONS.EQ.0) THEN
C TONNAGES FOR COMMENTS 1 AND 6 ARE APPROXIMATED
C IF (C.EQ.1) TONS=100
C IF (C.EQ.3) TONS=50
C IF (C.EQ.4) TONS=200
C IF (C.EQ.5) TONS=100
C IF (C.EQ.6) TONS=200
C IF (C.EQ.7) TONS=50
C IF (C.EQ.9) TONS=25
C END IF
C
C IF TONS ARE STILL ZERO, ADD ONE TO THE ZERO COUNTER.
C OTHERWISE, ADD THE TONS TO THE PROPER TONNAGE CLASS.
C
C TONS=TONS*W
C IF (TONS.EQ.0) THEN
C ZEROS=ZEROS+1
C N(1)=N(1)+1
C ELSE
C DO 310 I=1,3
C IF(TONS.LE.V(I)) THEN
C N(I)=N(I)+1
C GO TO 320
C END IF
310 CONTINUE
```



```
N(4)=N(4)+1  
END IF  
320 RETURN  
END
```